



TUGAS AKHIR - SM141501

PENEMUAN KEMBALI CITRA TENUN DENGAN KEMIRIPAN MOTIF MENGGUNAKAN TRANSFORMASI WAVELET DISKRIT

**AGISTA REGIASWURI
NRP 1212 100 028**

**Dosen Pembimbing
Dr. Dwi Ratna Sulistyaningrum, S.Si., M.T.**

**JURUSAN MATEMATIKA
Fakultas Matematika dan Ilmu Pengetahuan Alam
Institut Teknologi Sepuluh Nopember
Surabaya 2017**



FINAL PROJECT - SM141501

TENUN IMAGE RETRIEVAL WITH PATTERN SIMILARITY USING DISCRETE WAVELET TRANSFORM

**AGISTA REGIASWURI
NRP 1212 100 028**

**Supervisor
Dr. Dwi Ratna Sulistyaningrum, S.Si., M.T.**

**DEPARTMENT OF MATHEMATICS
Faculty of Mathematics and Natural Science
Sepuluh Nopember Institute of Technology
Surabaya 2017**

LEMBAR PENGESAHAN

PENEMUAN KEMBALI CITRA TENUN DENGAN KEMIRIPAN MOTIF MENGGUNAKAN TRANSFORMASI WAVELET DISKRIT

TENUN IMAGE RETRIEVAL WITH PATTERN SIMILARITY USING DISCRETE WAVELET TRANSFORM

TUGAS AKHIR

Diajukan untuk memenuhi salah satu syarat
Memperoleh gelar Sarjana Sains
Pada bidang minat Ilmu Komputer
Program Studi S-1 Jurusan Matematika
Fakultas Matematika dan Ilmu Pengetahuan Alam
Institut Teknologi Sepuluh Nopember Surabaya

Oleh :
AGISTA REGIASWURI
NRP. 1212 100 028

Menyetujui,
Dosen Pembimbing,



Dr. Dwi Ratna Sulistyaningrum, S.Si., M.T.
NIP. 19690405 199403 2 003

Mengetahui,
Ketua Jurusan Matematika
FMIPA ITS

Dr. Imam Mukhlash, S.Si., M.T.
NIP. 19700831 199403 1 003
Surabaya, 19 Januari 2017



PENEMUAN KEMBALI CITRA TENUN DENGAN KEMIRIPAN MOTIF MENGGUNAKAN TRANSFORMASI WAVELET DISKRIT

Nama Mahasiswa : Agista Regiaswuri
NRP : 1212 100 028
Jurusan : Matematika
Dosen Pembimbing : Dr. Dwi Ratna Sulistyaningrum,
S.Si., M.T.

Abstrak

Pencarian citra tenun pada suatu *database* citra biasanya dilakukan secara manual. Akibatnya, proses pencarian menjadi tidak efisien dan tidak efektif karena beberapa faktor seperti *keyword* yang terbatas, *human error* dan waktu pencarian yang lama. Tugas Akhir ini membuat aplikasi dengan kemampuan dapat mengekstraksi fitur tekstur citra tenun menggunakan Transformasi Wavelet Diskrit dan menemukan citra tenun dengan kemiripan motif menggunakan pengukuran kemiripan citra. Tahapan dari proses temu kembali citra tenun meliputi tahap baca citra, pra-pemrosesan, ekstraksi fitur tekstur menggunakan Transformasi Wavelet Diskrit dan pengukuran kemiripan citra menggunakan metode jarak Euclidean, jarak Manhattan dan jarak Canberra. Pengujian dilakukan pada 100 citra tenun, yang terdiri dari 80 citra di *database* dan 20 citra *query*. Proses yang dilakukan pada tahap pra-pemrosesan adalah *resize* citra dan *grayscale* citra. Proses ekstraksi menggunakan 4 jenis wavelet, yaitu Wavelet Haar, Wavelet Daubechies 4, Wavelet Symlets 2 dan Wavelet Coiflets 1. Hasil pengujian menunjukkan bahwa aplikasi dapat menemukan citra tenun yang memiliki kemiripan motif menggunakan Wavelet Haar dan jarak Manhattan dengan tingkat rata-rata persentase presisi tertinggi sebesar 85,41%.

Kata Kunci : Tenun, Temu Kembali Citra, Kemiripan Motif, Transformasi Wavelet Diskrit, Pengukuran Kemiripan.

TENUN IMAGE RETRIEVAL WITH PATTERN SIMILARITY USING DISCRETE WAVELET TRANSFORM

Name : ***Agista Regiaswuri***
NRP : ***1212 100 028***
Department : ***Mathematics***
Supervisor : ***Dr. Dwi Ratna Sulistyaningrum,***
S.Si., M.T.

Abstract

Tenun image search in an image database is usually done manually. As a result, the search process becomes inefficient and ineffective due to several factors such as keywords limited, human error and the long search time. This Final Project create application that can extract texture features of tenun image using Discrete Wavelet Transform and retrieve tenun images with pattern similarity using image similarity measurement. The stages of image retrieval process includes the step of reading image, pre-processing, feature extraction texture using Discrete Wavelet Transform and image similarity measurement using Euclidean distance, Manhattan distance and Canberra distance. Tests conducted on 100 tenun image, which consists of 80 images in the database and 20 query image. The pre-processing stage consists of image resizing and image grayscalling. The extraction process uses 4 types of wavelet. They are Haar, Daubechies 4, Symlets 2 and Coiflets 1. The test results showed that the application can retrieve the tenun images that has similar pattern using Haar wavelet and Manhattan distance with an average percentage rate of the highest precision amounting to 85.41%.

Keywords: *Tenun, Image Retrieval, Pattern Similarity, Discrete Wavelet Transform, Similarity Measurement.*

KATA PENGANTAR

Segala Puji bagi Allah SWT yang telah memberikan karunia, rahmat dan anugerah-Nya sehingga penulis dapat menyelesaikan Tugas Akhir berjudul: **“Penemuan Kembali Citra Tenun dengan Kemiripan Motif Menggunakan Transformasi Wavelet Diskrit”** yang merupakan salah satu persyaratan akademis dalam menyelesaikan Program Studi S-1 pada Jurusan Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Institut Teknologi Sepuluh Nopember Surabaya.

Tugas Akhir ini dapat diselesaikan dengan berkat kerjasama, bantuan dan dukungan dari banyak pihak. Sehubungan dengan hal itu, penulis mengucapkan terima kasih kepada:

1. Dr. Dwi Ratna Sulistyaningrum, S.Si., M.T. selaku dosen pembimbing yang senantiasa membimbing dengan sabar dan memberikan masukan dalam penyusunan Tugas Akhir.
2. Dr. Imam Mukhlash, S.Si., M.T. selaku Ketua Jurusan Matematika.
3. Dra. Wahyu Fistia Doctorina, M.Si. selaku dosen wali yang senantiasa membimbing dan menasehati penulis.
4. Drs. Nurul Hidayat, M. Kom., DR. Mahmud Yunus, M.Si., Sunarsini, S.Si., M.Si. dan Kistosil Fahim, S.Si., M.Si. selaku dosen penguji Tugas Akhir.
5. Dr. Didik Khusnul Arif, S.Si., M.Si. selaku Kaprodi Sarjana Matematika.
6. Drs. Iis Herisman, M.Sc. selaku Sekprodi Sarjana Matematika.
7. Seluruh jajaran dosen dan staf jurusan Matematika ITS.
8. Teman-teman mahasiswa jurusan Matematika ITS.

Penulis menyadari bahwa Tugas Akhir ini masih jauh dari kesempurnaan. Oleh karena itu, penulis mengharapkan saran dan kritik dari pembaca. Akhir kata, semoga Tugas Akhir ini bermanfaat bagi semua pihak yang berkepentingan.

Surabaya, Januari 2017

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
LEMBAR PENGESAHAN	v
ABSTRAK	vii
ABSTRACT	ix
KATA PENGANTAR	xi
DAFTAR ISI	xiii
DAFTAR GAMBAR	xvii
DAFTAR TABEL	xix
DAFTAR LAMPIRAN	xxi
 BAB I. PENDAHULUAN	
1.1 Latar Belakang Masalah	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah	3
1.4 Tujuan	3
1.5 Manfaat	4
1.6 Sistematika Penulisan	4
 BAB II. TINJAUAN PUSTAKA	
2.1 Penelitian Sebelumnya	7
2.2 Teori Penunjang	8
2.2.1 Tenun	8
2.2.2 <i>Content Based Image Retrieval</i> (CBIR)	10
2.2.3 Tekstur	13
2.2.4 Transformasi Wavelet	14
2.2.5 Transformasi Wavelet Diskrit	16
2.2.6 Representasi Fitur Citra	19
2.2.7 Pengukuran Kemiripan Citra	20
 BAB III. METODOLOGI	
3.1 Objek Penelitian	23
3.2 Peralatan Penelitain	23
3.3 Tahapan Penelitian	24

3.4 Diagram Blok	25
BAB IV. PERANCANGAN DAN IMPLEMENTASI	
4.1 Perancangan Aplikasi	29
4.1.1 Perancangan Data	29
4.1.2 Gambaran Aplikasi Secara Umum	31
4.1.3 Tahap Baca Citra Tenun	35
4.1.4 Tahap Pra-Pemrosesan Citra Tenun.....	35
4.1.5 Tahap Ekstraksi Fitur Tekstur Menggunakan Transformasi Wavelet Diskrit	36
4.1.6 Tahap Pengukuran Kemiripan Citra Tenun	37
4.2 Implementasi	38
4.2.1 Implementasi Antarmuka	38
4.2.2 Implementasi Tahap Baca Citra Tenun ..	43
4.2.3 Implementasi Tahap Pra-Pemrosesan Citra Tenun	43
4.2.4 Implementasi Tahap Ekstraksi Fitur Tekstur Menggunakan Transformasi Wavelet Diskrit	44
4.2.5 Implementasi Tahap Pengukuran Kemiripan Citra Tenun	45
BAB V. PENGUJIAN DAN PEMBAHASAN HASIL	
5.1 Pengujian Aplikasi	47
5.1.1 Lingkungan Pengujian Aplikasi	47
5.1.2 Pengujian Tahap Baca Citra Tenun	48
5.1.3 Pengujian Tahap Pra-Pemrosesan Citra Tenun	49
5.1.4 Pengujian Tahap Ekstraksi Fitur Tekstur Menggunakan Transformasi Wavelet Diskrit	49
5.1.5 Pengujian Tahap Pengukuran Kemiripan Citra Tenun	51
5.2 Pembahasan Hasil	58

BAB VI. PENUTUP

6.1 Simpulan	61
6.2 Saran	62

DAFTAR PUSTAKA	63
-----------------------------	-----------

LAMPIRAN	65
-----------------------	-----------

DAFTAR GAMBAR

	Halaman
Gambar 2.1 <i>Content Based Image Retrieval System</i>	12
Gambar 2.2 Tekstur Berdasarkan Struktur	14
Gambar 2.3 Dekomposisi Wavelet Citra Dua Dimensi	16
Gambar 2.4 Dekomposisi Wavelet sampai Level 2	17
Gambar 2.5 Jenis Wavelet	18
Gambar 3.1 Diagram Blok Proses Temu Kembali Citra Tenun	27
Gambar 4.1 Diagram Alir Proses Temu Kembali pada Citra Tenun di <i>Database</i>	33
Gambar 4.2 Diagram Alir Proses Temu Kembali Citra Tenun	34
Gambar 4.3 Antarmuka Utama	38
Gambar 4.4 Antarmuka Proses dan Uji	40
Gambar 5.1 Hasil Baca Citra Tenun	48
Gambar 5.2 Hasil Pra-Pemrosesan <i>Resize</i> dan <i>Grayscale</i>	49
Gambar 5.3 Contoh Citra dengan Kualitas Rendah	59

DAFTAR TABEL

	Halaman
Tabel 2.1 Ragam Motif Tenun	9
Tabel 2.2 Nilai Koefisien Wavelet	19
Tabel 4.1 Data Proses	30
Tabel 4.2 Kegunaan Menu Aplikasi	39
Tabel 5.1 Lingkungan Pengujian Aplikasi	47
Tabel 5.2 Kelompok Citra Tenun	48
Tabel 5.3 Hasil Dekomposisi Wavelet Level 1 sampai Level 4	50
Tabel 5.4 Hasil Ekstraksi Fitur Tekstur Citra di <i>Database</i> Menggunakan Transformasi Wavelet Diskrit.....	51
Tabel 5.5 Hasil Pengukuran Kemiripan Citra dengan Mengatur Jumlah yang Ingin Ditampilkan.....	52
Tabel 5.6 Hasil Pengukuran Kemiripan Citra dengan Mengatur Nilai <i>Threshold</i>	53
Tabel 5.7 Hasil Pengukuran Kemiripan Citra Menggunakan Jarak Euclidean	55
Tabel 5.8 Hasil Pengukuran Kemiripan Citra Menggunakan Jarak Manhattan	56
Tabel 5.9 Hasil Pengukuran Kemiripan Citra Menggunakan Jarak Canberra	57
Tabel 5.10 Perbandingan Presisi dari Dekomposisi Level 1 sampai Level 4	58

DAFTAR LAMPIRAN

	Halaman
LAMPIRAN A	65
A.1 Kode Fungsi untuk Ekstraksi Fitur Tekstur Menggunakan Wavelet Haar	65
A.2 Kode Fungsi untuk Ekstraksi Fitur Tekstur Menggunakan Wavelet Daubechies 4	66
A.3 Kode Fungsi untuk Ekstraksi Fitur Tekstur Menggunakan Wavelet Symlets 2	67
A.4 Kode Fungsi untuk Ekstraksi Fitur Tekstur Menggunakan Wavelet Coiflets 1	68
LAMPIRAN B	69
B.1 Kode Fungsi Pengukuran Kemiripan Citra dengan Mengatur Jumlah Citra untuk Jarak Euclidean	69
B.2 Kode Fungsi Pengukuran Kemiripan Citra dengan Mengatur Jumlah Citra untuk Jarak Manhattan	71
B.3 Kode Fungsi Pengukuran Kemiripan Citra dengan Mengatur Jumlah Citra untuk Jarak Canberra	73
B.4 Kode Fungsi Pengukuran Kemiripan Citra dengan Mengatur Nilai <i>Threshold</i> untuk Jarak Euclidean	75
B.5 Kode Fungsi Pengukuran Kemiripan Citra dengan Mengatur Nilai <i>Threshold</i> untuk Jarak Manhattan	77
B.6 Kode Fungsi Pengukuran Kemiripan Citra dengan Mengatur Nilai <i>Threshold</i> untuk Jarak Canberra	79

BAB I

PENDAHULUAN

Bab ini menguraikan hal-hal yang menjadi latar belakang Tugas Akhir untuk selanjutnya dituliskan dalam subbab perumusan masalah. Bab ini juga mencantumkan batasan masalah, tujuan dan manfaat Tugas Akhir. Adapun sistematika penulisan Tugas Akhir diuraikan pada bagian akhir bab.

1.1 Latar Belakang Masalah

Tenun sebagai salah satu warisan budaya merupakan suatu kebanggaan bagi bangsa Indonesia. Tenun memiliki motif atau corak yang beraneka ragam sesuai dengan kultur masyarakat Indonesia yang juga beragam. Beberapa daerah di Indonesia yang terkenal dengan tenun diantaranya adalah Toraja, Sintang, Jepara, Bali, Lombok, Sumbawa, Sumba, dan Flores. Perbedaan motif khas tenun dari masing-masing daerah sangat dipengaruhi oleh letak geografis kepulauan, keadaan alam dan struktur masyarakat. Ragam hias yang terdapat dalam motif tenun seperti flora, fauna dan geometris menjadi suatu keunikan yang membedakan setiap nama, daerah asal dan filosofi suatu motif.

Apabila seluruh data tentang tenun hanya tersimpan secara konvensional, maka penyimpanan data yang berupa citra tenun tersebut menjadi terbatas dan tidak permanen. Selain itu, proses untuk melakukan pencarian kembali suatu citra akan rumit sehingga mengakibatkan proses pencarian citra tenun menjadi tidak efisien dan tidak efektif. Permasalahan tentang cara menyimpan data citra tenun dalam jumlah besar telah memiliki solusi, salah satunya adalah penggunaan *database*. Namun, karena jumlah data terus bertambah, saat ini permasalahan berkembang ke arah cara pencarian citra tenun di dalam suatu kumpulan data citra yang besar tersebut.

Di sisi lain, teknologi dengan pemanfaatan metode temu kembali atau sering disebut “*retrieval*” semakin berkembang. Metode temu kembali berkaitan dengan pengindeksan dan pengambilan informasi dari sumber informasi yang heterogen. Contoh penerapan

metode temu kembali dalam kehidupan sehari-hari yang sering ditemui adalah penggunaan mesin pencari. Mesin pencari digunakan untuk mencari informasi yang relevan sesuai dengan kata kunci (*keyword*) yang telah dimasukkan oleh pengguna.

Saat ini, metode temu kembali tidak hanya berbasis teks, tetapi juga berbasis isi visual dari suatu citra. Metode temu kembali citra berbasis isi atau *Content Based Image Retrieval* (CBIR) merupakan metode pencarian citra dengan cara membandingkan citra contoh (*query image*) dan citra di dalam *database* (*training image*). Salah satu kegunaan CBIR adalah menemukan data citra berdasarkan fitur-fitur penting yang merupakan karakteristik atau ciri dari suatu citra sehingga dapat diidentifikasi secara akurat. CBIR banyak dimanfaatkan dalam berbagai bidang, seperti pencocokan sidik jari, pengindraan jarak jauh, deteksi tumor dan lain sebagainya [1].

Secara umum, metode CBIR diimplementasikan dengan melihat fitur dasar citra, seperti warna, tekstur dan bentuk. Namun, tekstur merupakan fitur terpenting pada suatu citra karena cukup banyak citra yang dapat diidentifikasi dengan komposisi tekstur yang berbeda [2]. Selain itu, tekstur juga merupakan fitur penting pada citra tenun. Hal tersebut dapat dilihat pada motif tenun yang membentuk pengulangan warna ataupun perubahan warna.

Penelitian yang mengangkat tenun sebagai studi kasus telah dilakukan oleh N. M. Setiohardjo dan A. Harjoko (2014), yaitu tentang analisis tekstur untuk klasifikasi motif kain tenun [3]. Penelitian tersebut menggunakan metode *Gray Level Co-occurrence Matrix* (GLCM) dan *Color Co-occurrence Matrix* (CCM) untuk menganalisis tekstur tenun. Selain itu, penelitian tentang metode temu kembali citra berbasis isi telah dilakukan oleh S. Bharkad dan M. Kokare (2013), yaitu pencocokan sidik jari menggunakan transformasi paket wavelet diskrit [4]. Penelitian tersebut menggunakan transformasi wavelet untuk ekstraksi fitur tekstur pada citra sidik jari. Penelitian lainnya adalah tentang temu kembali citra medis menggunakan *Fuzzy C-Means Clustering* [5]. Penelitian yang berbasis pada fitur bentuk dan tekstur tersebut dilakukan oleh J. Samraj dan M. NazreenBee (2015).

Berdasarkan data dan fakta yang telah disebutkan, perkembangan teknologi dengan pemanfaatan metode pencarian citra berbasis isi dapat digunakan untuk mengatasi permasalahan pencarian citra tenun pada suatu *database*. Selain itu, transformasi wavelet dapat digunakan untuk mengekstraksi fitur tekstur, khususnya pada citra tenun. Oleh karena itu, perlu dilakukan penelitian tentang “Penemuan Kembali Citra Tenun dengan Kemiripan Motif Menggunakan Transformasi Wavelet Diskrit” sehingga dapat dirancang suatu aplikasi Temu Kembali Citra Tenun berbasis fitur tekstur.

1.2 Rumusan Masalah

Permasalahan yang dihadapi pada Tugas Akhir dapat dirumuskan sebagai berikut:

1. Bagaimana mengekstraksi fitur tekstur pada citra tenun menggunakan Transformasi Wavelet Diskrit?
2. Bagaimana menemukan citra tenun yang memiliki kemiripan motif menggunakan pengukuran kemiripan citra?

1.3 Batasan Masalah

Batasan-batasan masalah yang berguna untuk memperjelas permasalahan yang diteliti pada Tugas Akhir diantaranya sebagai berikut:

1. *Software* yang digunakan untuk membuat aplikasi Tugas Akhir adalah MATLAB.
2. Citra yang digunakan berupa citra tenun berukuran 300x300 piksel dengan format JPG atau JPEG.
3. Citra tenun diperoleh dari pengambilan gambar pada kain tenun dengan beberapa motif.
4. Jenis motif merupakan motif tenun khas Bali, Jawa Tengah, Nusa Tenggara Timur dan Nusa Tenggara Barat.

1.4 Tujuan

Tujuan dari pengerjaan Tugas Akhir adalah membuat aplikasi dengan kemampuan dapat mengekstraksi fitur tekstur citra tenun menggunakan Transformasi Wavelet Diskrit dan menemukan citra

tenun dengan kemiripan motif dengan cara mengukur kemiripan fitur hasil ekstraksi menggunakan pengukuran kemiripan citra.

1.5 Manfaat

Manfaat yang dapat diperoleh dari Tugas Akhir adalah memberikan solusi alternatif dalam hal pengelolaan dan pencarian data citra, khususnya citra tenun, sebagai upaya untuk mengenalkan tenun ke masyarakat luas sekaligus mendukung konsep pemasaran *e-Commerce*. Selain itu, penelitian ini juga mendukung konsep mesin pencari berbasis gambar.

1.6 Sistematika Penulisan

Sistematika penulisan Tugas Akhir disusun sebagai berikut:

1. Bab 1 Pendahuluan
Bab ini menjelaskan tentang latar belakang dari pengerjaan Tugas Akhir, rumusan masalah yang dihadapi dalam pengerjaan Tugas Akhir, batasan masalah yang digunakan selama pengerjaan Tugas Akhir, tujuan dan manfaat Tugas Akhir, serta sistematika penulisan Tugas Akhir.
2. Bab 2 Tinjauan Pustaka
Bab ini menjelaskan tentang penelitian-penelitian sebelumnya yang mendukung Tugas Akhir dan beberapa teori penunjang yang digunakan untuk membantu proses pengerjaan Tugas Akhir.
3. Bab 3 Metodologi
Bab ini menjelaskan tentang metode yang digunakan untuk melakukan ekstraksi fitur tekstur citra tenun hingga menemukan citra dengan kemiripan motif menggunakan Transformasi Wavelet Diskrit, diantaranya adalah objek penelitian, peralatan penelitian dan tahapan penelitian.

4. Bab 4 Perancangan dan Implementasi
Bab ini menjelaskan tentang perancangan aplikasi dan implementasinya, sehingga dapat diketahui tahap pembuatan aplikasi secara utuh beserta cara untuk menemukan citra tenun dengan kemiripan motif.
5. Bab 5 Pengujian dan Pembahasan Hasil
Bab ini menjelaskan tentang pengujian aplikasi pada setiap tahap dalam menemukan citra tenun dengan kemiripan motif beserta pembahasan terhadap hasil pengujian yang telah diperoleh.
6. Bab 6 Simpulan dan Saran
Bab ini merupakan penutup, berisi tentang simpulan yang dapat diambil berdasarkan hasil data yang diperoleh dan saran yang selayaknya dilakukan apabila Tugas Akhir akan dikembangkan.

BAB II

TINJAUAN PUSTAKA

Bab ini menjelaskan tentang penelitian-penelitian sebelumnya yang mendukung Tugas Akhir dan teori-teori penunjang yang digunakan dalam pengerjaan Tugas Akhir, diantaranya adalah tentang tenun, *Content Based Image Retrieval* (CBIR), tekstur, transformasi wavelet, Transformasi Wavelet Diskrit atau *Discrete Wavelet Transform* (DWT), representasi fitur citra, dan pengukuran kemiripan citra.

2.1 Penelitian Sebelumnya

Penelitian sebelumnya yang berkaitan dengan Tugas Akhir salah satunya adalah “Analisis Tekstur untuk Klasifikasi Motif Kain (Studi Kasus Kain Tenun Nusa Tenggara Timur)” [3]. Penelitian tersebut dilakukan oleh Nicodemus Mardanus Setiohardjo dan Agus Harjoko dengan tujuan membandingkan metode GLCM dan CCM yang digunakan untuk menganalisis tekstur pada citra tenun. Selanjutnya, hasil dari proses ekstraksi fitur akan digunakan sebagai masukan untuk proses klasifikasi.

Sangita Bharkad dan Manesh Kokare melakukan penelitian dengan judul “*Fingerprint Matching Using Discrete Wavelet Packet Transform*” [4]. Penelitian tersebut menggunakan metode *Discrete Wavelet Packet Transform* (DWPT) untuk mengekstraksi fitur tekstur pada citra sidik jari. Hasil dari penelitian tersebut dapat dimanfaatkan pada sistem verifikasi sidik jari.

Jasmine Samraj dan NazreenBee M. melakukan penelitian dengan judul “*Content Based Medical Image Retrieval Using Fuzzy C-Means Clustering With RF*” [5]. Penelitian tersebut mengekstraksi fitur bentuk dan fitur tekstur citra hasil *scan* MRI. Hasil dari penelitian tersebut dapat dimanfaatkan pada sistem deteksi tumor.

Berdasarkan beberapa penelitian yang telah dilakukan sebelumnya, penelitian yang dilakukan pada Tugas Akhir ini adalah tentang identifikasi citra tenun berbasis isi visual citra sehingga dapat dirancang suatu aplikasi Temu Kembali Citra Tenun berbasis fitur

tekstur. Aplikasi dirancang menggunakan metode Transformasi Wavelet Diskrit atau *Discrete Wavelet Transform* (DWT) untuk ekstraksi fitur tekstur. Kemudian, kemiripan citra diukur menggunakan metode perhitungan jarak.

2.2 Teori Penunjang

2.2.1 Tenun

Teknik tenun merupakan teknik dalam pembuatan kain yang dibuat dengan prinsip sederhana, yaitu menggabungkan benang secara memanjang dan melintang [6]. Benang yang terletak secara vertikal atau memanjang pada alat tenun disebut benang lungsin, sedangkan benang yang keluar masuk pada lungsi secara horizontal atau mengikuti lebar kain disebut benang pakan. Benang-benang tersebut ditenun sedemikian rupa sehingga menghasilkan beraneka ragam motif sebagai akibat dari paduan warna benang.

Kain tenun terbuat dari serat kayu, kapas atau sutra. Sebuah kain tenun melalui proses pembuatan yang panjang, yaitu dari kapas yang dicelupkan ke pewarna, kemudian diikat dan dirajut secara selang-seling menjadi sebuah benang hingga helaian kain. Berdasarkan teknik pembuatannya, terdapat beberapa jenis hasil tenun yaitu tenun sederhana, ikat lungsi, ikat pakan, ikat ganda, songket dan *dobby*. Kualitas hasil tenun sering dilihat dari mutu bahan, keindahan tata warna, motif dan ragam hiasannya. Produksi kain tenun biasanya dilakukan dalam skala rumah tangga.

Seni tenun berkaitan erat dengan ilmu pengetahuan, budaya, kepercayaan, lingkungan alam atau sistem organisasi sosial dari suatu masyarakat yang memiliki budaya tenun tersebut. Karena masyarakat Indonesia memiliki kultur sosial yang beragam, maka seni tenun pada setiap daerah selalu bersifat partikular atau memiliki ciri khas yang merupakan representasi dari masing-masing budaya masyarakat. Ragam hias yang terdapat dalam motif tenun seperti flora, fauna dan geometris menjadi suatu keunikan yang membedakan setiap nama, daerah asal dan filosofi suatu motif.

Beberapa daerah yang terkenal dengan produksi tenunnya adalah Jawa Tengah, Bali dan Nusa Tenggara Timur. Motif tenun yang terkenal dan banyak diproduksi dari masing-masing daerah

tersebut diantaranya adalah motif rang-rang dari Jawa Tengah, motif endek dari Bali, motif lurik dari Nusa Tenggara Timur dan motif Sikka dari Nusa Tenggara Barat. Ragam motif tenun dari beberapa daerah di Indonesia disajikan pada Tabel 2.1.

Tabel 2.1 Ragam Motif Tenun [6]

Daerah Asal	Nama Motif	Gambar Tenun
Bali	Endek	
Jawa Tengah	Rang-Rang	
NTT	Lurik	
NTB	Sikka	

Motif rang-rang konon memiliki arti jarang-jarang atau berlubang. Jarang-jarang maksudnya adalah motif antara warna yang satu dengan warna yang lain saling berjauhan sehingga menampilkan kesan tegas, sedangkan berlubang maksudnya adalah jumlah benang yang digunakan lebih sedikit sehingga tenun terkesan lebih renggang. Selain memiliki ciri khas warna yang tegas, motif rang-rang juga memiliki ciri garis yang tegas dan sederhana menyerupai zig-zag.

Motif endek merupakan tenun yang motifnya dibuat dengan teknik nyantri, yaitu memberikan coretan pada bagian-bagian tertentu sehingga menyerupai teknik melukis. Ciri khas dari motif endek adalah memiliki perpaduan warna yang kaya dan menyerupai bagian tumbuh-tumbuhan seperti bunga atau dedaunan.

Motif lurik memiliki motif garis-garis kecil searah panjang kain. Selain itu, motif lurik juga memiliki kombinasi warna yang sederhana, yaitu warna-warna yang senada. Sedangkan motif sikka memiliki motif fauna seperti cicak atau biawak. Hewan-hewan tersebut dianggap sakral dan melambangkan kehidupan di dunia bawah.

2.2.2 Content Based Image Retrieval (CBIR)

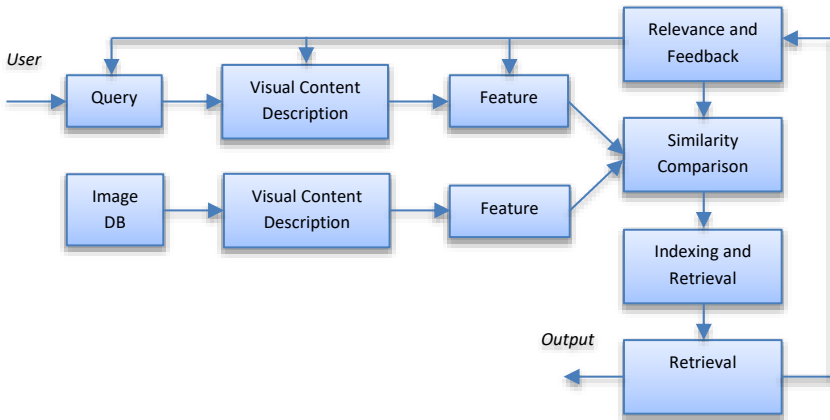
Setiap citra memiliki ciri atau karakteristik yang berbeda. Ciri merupakan suatu tanda khas yang menjadi pembeda dari suatu citra dengan citra lainnya. Sebagai contoh, bunga matahari dengan bunga melati dapat dibedakan melalui warnanya, kain dengan kertas dapat dibedakan melalui teksturnya, dan suatu objek yang berbentuk lingkaran dengan objek kotak dapat dibedakan melalui bentuknya. Ciri yang menonjol dari suatu citra dapat diperoleh melalui proses ekstraksi fitur. Namun, suatu citra dapat memiliki *multiple feature* sehingga mengakibatkan proses ekstraksi menjadi tidak mudah.

Content Based Image Retrieval (CBIR) merupakan salah satu metode untuk mencari suatu citra dengan cara membandingkan citra *query* dan setiap citra di *database*. Kelebihan utama dari metode tersebut adalah kemampuannya untuk mendukung *query* visual. Secara prinsip, cara kerja metode CBIR berbeda dengan metode pencarian citra menggunakan kata kunci.

Sistem *Content Based Image Retrieval* (CBIR) dapat dibangun berdasarkan karakteristik atau ciri suatu citra. Secara umum, suatu citra memiliki beberapa fitur dasar sebagai berikut:

1. Warna
Warna merupakan spektrum cahaya yang dipantulkan suatu benda dan ditangkap oleh sistem *visual* manusia. Warna yang diterima oleh mata memiliki panjang gelombang yang berbeda-beda. Kombinasi warna dengan rentang terlebar adalah *Red* (R), *Green* (G) dan *Blue* (B). Ketiga warna tersebut merupakan warna primer dan disingkat RGB.
2. Bentuk
Bentuk mencerminkan konfigurasi atau kerangka objek. Secara umum, citra yang dibentuk oleh mata merupakan citra dua dimensi, sedangkan objek yang dilihat umumnya berbentuk tiga dimensi. Bentuk merupakan atribut yang jelas sehingga dapat digunakan untuk mendeteksi banyak objek atau batas wilayah dengan mengenali berdasarkan bentuknya saja, seperti bentuk lingkaran atau persegi.
3. Tekstur
Tekstur adalah frekuensi perubahan atau pengulangan warna pada suatu citra. Tekstur menunjukkan distribusi spasial dari derajat keabuan pada kumpulan piksel, sehingga *visual* manusia tidak dapat menerima informasi citra dari setiap piksel, melainkan sebagai satu kesatuan.

Content Based Image Retrieval (CBIR) dapat menggunakan salah satu maupun kombinasi dari beberapa fitur tersebut untuk menunjukkan citra yang dicari sesuai dengan keinginan pengguna. Sistem CBIR dapat dilihat pada Gambar 2.1.



Gambar 2.1 *Content Based Image Retrieval System* [7]

Content Based Image Retrieval (CBIR) mengekstraksi fitur visual suatu citra dan menggambarkan hasilnya sebagai vektor fitur multi-dimensional, sehingga vektor fitur yang berada di dalam basis data membentuk sebuah *dataset* fitur. Pengguna memberikan contoh citra (citra *query*) pada sistem. Kemudian, sistem menghitung nilai-nilai vektor fitur pada citra *query* tersebut. Selanjutnya, tingkat kemiripan yang berupa besar-kecilnya nilai jarak antara vektor fitur citra *query* di dalam basis data dihitung dan diurutkan dari nilai terkecil. Proses pengurutan jarak tersebut akan mempermudah pengguna dalam mencari citra yang tersimpan di dalam *database*.

Content Based Image Retrieval (CBIR) dapat diimplementasikan dalam berbagai bidang, beberapa diantaranya adalah sebagai berikut [1]:

1. Sistem pengenalan wajah untuk pencegahan tindak kriminal.
2. Perlindungan hak cipta dalam bidang jurnalistik.
3. Sistem pencocokan sidik jari dalam bidang forensik.
4. Pendeteksi tumor dalam bidang medis.
5. Pengindraan jarak jauh berupa sistem informasi ramalan cuaca.

2.2.3 Tekstur

Tekstur merupakan ciri utama atau fitur utama yang sering digunakan untuk menganalisis suatu objek. Tekstur berkaitan dengan tingkat kekasaran (*roughness*), granularitas (*granulation*) dan keteraturan (*regularity*) susunan struktural piksel. Tekstur memiliki frekuensi perubahan warna pada citra yang dinyatakan dalam bentuk kasar, sedang dan halus. Misalnya, pada kasus pengenalan objek citra untuk penginderaan jarak jauh, hutan bertekstur kasar, belukar bertekstur sedang dan semak bertekstur halus. Tekstur dapat didefinisikan sebagai suatu keteraturan pola-pola tertentu yang terbentuk dari susunan piksel-piksel dalam citra digital. Artinya, suatu permukaan dalam *image grayscale* dikatakan memiliki informasi tekstur apabila muncul pola-pola teratur dalam interval jarak dan arah tertentu secara berulang-ulang. Sehingga, satu piksel saja tidak dapat didefinisikan sebagai tekstur.

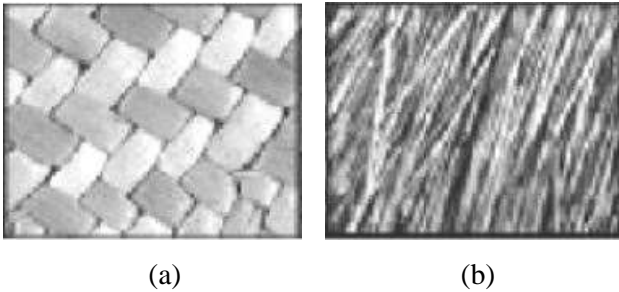
Berdasarkan strukturnya, tekstur dapat diklasifikasikan ke dalam dua golongan, yaitu:

1. Makrostruktur

Tekstur makrostruktur memiliki perulangan pola lokal secara periodik pada suatu daerah citra, biasanya terdapat pada pola-pola buatan manusia dan cenderung mudah untuk direpresentasikan secara matematis.

2. Mikrostruktur

Pola-pola lokal dan perulangan pada tekstur mikrostruktur tidak terjadi begitu jelas, sehingga tidak mudah untuk memberikan definisi tekstur yang komprehensif.



Gambar 2.2 Tekstur Berdasarkan Struktur [8].

(a) Tekstur Makrostruktur; (b) Tekstur Mikrostruktur.

Tekstur memiliki peran penting pada banyak aplikasi pengolahan citra, seperti pengindraan jarak jauh dan pencitraan medis. Sehingga, diperlukan analisis mengenai tekstur untuk mendeskripsikan suatu tekstur agar dapat digunakan untuk mengubah, membandingkan dan mentransformasikan tekstur. Algoritma analisis tekstur cenderung melakukan proses ekstraksi fitur dan merepresentasikan hasil ekstraksi ke dalam skema pengkodean citra. Beberapa metode yang biasa digunakan untuk menganalisis tekstur diantaranya adalah metode Transformasi Wavelet, metode *Gabor Filter* dan metode GLCM (*Gray Level Co-occurrence Matrix*). Metode-metode tersebut pernah digunakan untuk menganalisis tekstur pada citra batik, citra medis dan citra sidik jari.

2.2.4 Transformasi Wavelet

Wavelet merupakan suatu gelombang kecil, sedangkan sinus dan cosinus merupakan gelombang besar. Wavelet adalah fungsi yang memenuhi persyaratan matematika tertentu sehingga mampu melakukan dekomposisi terhadap sebuah fungsi secara hirarki. Wavelet dapat digunakan untuk menggambarkan sebuah model atau gambar asli berupa citra, kurva atau sebuah bidang ke dalam fungsi matematis.

Transformasi wavelet adalah suatu fungsi konversi yang dapat membagi fungsi atau sinyal ke dalam komponen frekuensi atau skala yang berbeda sehingga komponen-komponen tersebut dapat dipelajari dengan resolusi tertentu sesuai dengan skalanya.

Wavelet merupakan sebuah fungsi variabel real x dalam ruang fungsi $L^2(R)$. Fungsi tersebut berasal dari sebuah *scaling function* yang memiliki sifat dapat disusun dari sejumlah salinan dirinya yang telah didilasikan, ditranslasikan dan diskalakan. *Scaling function* dinyatakan sebagai berikut [9]:

$$\phi(x) = 2 \sum_k c_k \phi(2x - k); \quad k \in \mathbb{Z} \quad (2.1)$$

dengan c adalah koefisien transformasi atau koefisien tapis (*filter*).

Sedangkan persamaan wavelet pertama yang disebut sebagai *mother wavelet* dinyatakan sebagai berikut [9]:

$$\psi^0(x) = 2 \sum_k c_0 k \psi(2x - k); \quad k \in \mathbb{Z} \quad (2.2)$$

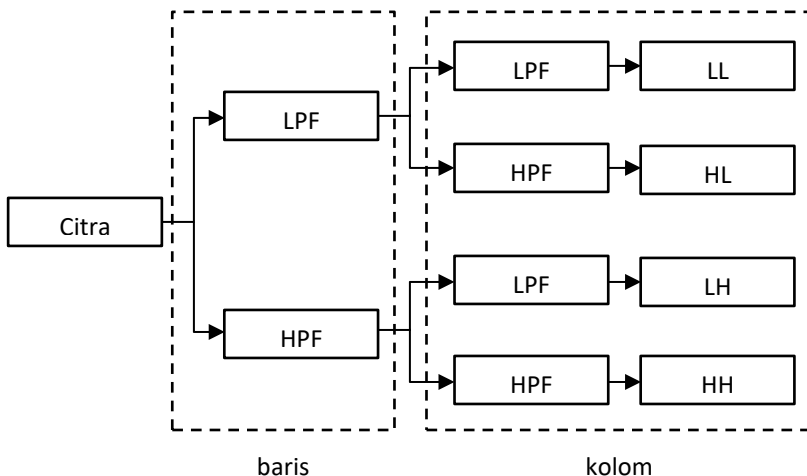
dengan c_0 adalah koefisien wavelet. *Mother wavelet* tersebut dapat digunakan untuk membentuk wavelet-wavelet lainnya ($\psi^0, \psi^1, \psi^2, \dots$) dengan cara didilasikan (dimampatkan atau diregangkan) dan ditranslasikan (digeser).

Transformasi wavelet dibedakan berdasarkan nilai parameter dilasi dan translasinya, yaitu *Continuous Wavelet Transform* (CWT) atau Transformasi Wavelet Kontinu dan *Discrete Wavelet Transform* (DWT) atau Transformasi Wavelet Diskrit. Transformasi Wavelet Kontinu ditentukan oleh nilai parameter dilasi dan nilai parameter translasi yang bervariasi secara kontinu. Transformasi Wavelet Diskrit didefinisikan untuk mengurangi redundansi yang terjadi pada transformasi kontinu dengan hanya mengambil nilai diskrit dari parameter dilasi dan translasi.

2.2.5 Transformasi Wavelet Diskrit

Transformasi wavelet merepresentasikan sebarang fungsi sebagai superposisi wavelet. Setiap superposisi mendekomposisi fungsi tersebut ke level skala yang berbeda, kemudian setiap level tersebut didekomposisi lebih lanjut. Transformasi Wavelet Diskrit merupakan suatu transformasi sinyal diskrit menjadi koefisien-koefisien wavelet dengan cara menapis sinyal menggunakan dua buah tapis yang berlawanan, yaitu *Low Pass Filter* (LPF) dan *High Pass Filter* (HPF).

Discrete Wavelet Transform (DWT) berkaitan dengan sistem *sub-band* hirarki. *Sub-band* merupakan jarak logaritmis dalam domain frekuensi. Citra dua dimensi memiliki prosedur dekomposisi level tunggal yang terdiri dari citra satu dimensi yang di-*filter* pada arah mendatar, kemudian diikuti oleh citra satu dimensi yang di-*filter* pada arah tegak dan diutilisasi menggunakan LPF dan HPF. Proses dekomposisi wavelet terhadap citra dua dimensi dapat dilihat pada Gambar 2.3.

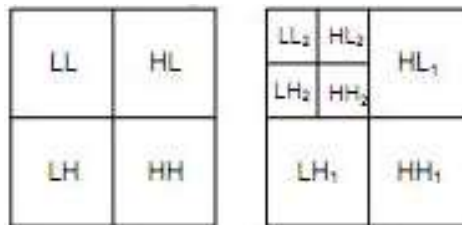


Gambar 2.3 Dekomposisi Wavelet Citra Dua Dimensi [10]

Algoritma DWT mendekomposisi sebuah dimensi sinyal menjadi dua bagian, yaitu bagian dengan frekuensi tinggi dan bagian dengan frekuensi rendah. Sebuah sinyal dilewatkan melalui HPF untuk menganalisis frekuensi tinggi atau bagian detail, kemudian juga dilewatkan melalui LPF untuk menganalisis frekuensi rendah atau bagian *smooth*. Hasil dari penapisan citra menggunakan transformasi wavelet adalah 4 blok atau *sub-band* citra, yaitu sebagai berikut:

1. LL atau aproksimasi : LPF terhadap baris, kemudian LPF terhadap kolom.
2. LH atau detail horizontal : LPF terhadap baris, kemudian HPF terhadap kolom.
3. HL atau detail vertikal : HPF terhadap baris, kemudian LPF terhadap kolom.
4. HH atau detail diagonal : HPF terhadap baris, kemudian HPF terhadap kolom.

Masing-masing *sub-band* tersebut berada dalam kawasan wavelet. *Sub-band* frekuensi rendah (LL) mengacu pada fitur tekstur, sedangkan *sub-band* yang lain mengacu ada informasi tepi dalam orientasi vertikal, horizontal dan diagonal.

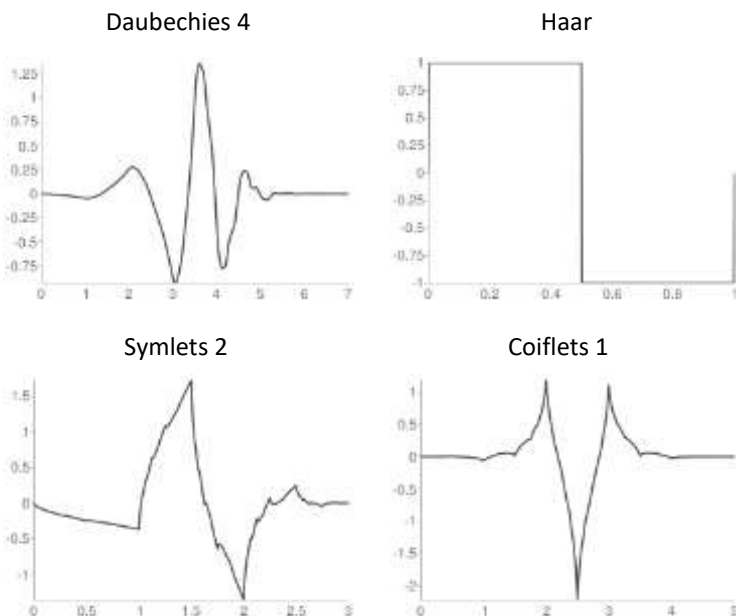


Gambar 2.4 Dekomposisi Wavelet sampai Level 2 [11]

Aproksimasi koefisien wavelet pada level berikutnya dapat diperoleh dengan melakukan dekomposisi lebih lanjut pada *sub-band* LL dari level sebelumnya. Gambar 2.4 menunjukkan hasil dekomposisi wavelet sampai level 2. Pada level 2, koefisien wavelet

diperoleh dengan mendekomposisi *sub-band* LL pada level 1 (LL1). Sedangkan pada level 3, koefisien wavelet diperoleh dengan mendekomposisi *sub-band* LL1 pada level 2 (LL2). Demikian juga untuk level-level selanjutnya, dapat dilakukan dekomposisi sampai level terakhir yang ingin dicapai.

Jenis wavelet bervariasi, beberapa diantaranya adalah Wavelet Daubechies 4, Wavelet Haar, Wavelet Symlets 2 dan Wavelet Coiflets 1 dapat dilihat pada Gambar 2.5.



Gambar 2.5 Jenis Wavelet [12]

Masing-masing wavelet tersebut dibedakan oleh nilai-nilai koefisien yang ditunjukkan pada Tabel 2.2.

Tabel 2.2 Nilai Koefisien Wavelet [12]

Jenis Wavelet	Dekomposisi LPF	Dekomposisi HPF
Haar	$c_0 = 0.7071067812$ $c_1 = 0.7071067812$	$d_0 = -0.7071067812$ $d_1 = 0.7071067812$
Daubechies 4	$c_0 = -0.0105974018$ $c_1 = 0.0328830117$ $c_2 = 0.0308413818$ $c_3 = -0.1870348117$ $c_4 = -0.0279837694$ $c_5 = 0.6308807679$ $c_6 = 0.7148465706$ $c_7 = 0.2303778133$	$d_0 = -0.2303778133$ $d_1 = 0.7148465706$ $d_2 = -0.6308807679$ $d_3 = -0.0279837694$ $d_4 = 0.1870348117$ $d_5 = 0.0308413818$ $d_6 = -0.0328830117$ $d_7 = -0.0105974018$
Symlets 2	$c_0 = -0.1294095226$ $c_1 = 0.2241438680$ $c_2 = 0.8365163037$ $c_3 = 0.4829629131$	$d_0 = -0.4829629131$ $d_1 = 0.8365163037$ $d_2 = -0.2241438680$ $d_3 = -0.1294095226$
Coiflets 1	$c_0 = -0.0156557281$ $c_1 = -0.0727326195$ $c_2 = 0.3848648469$ $c_3 = 0.8525720202$ $c_4 = 0.3378976625$ $c_5 = -0.0727326195$	$d_0 = 0.0727326195$ $d_1 = 0.3378976625$ $d_2 = -0.8525720202$ $d_3 = 0.3848648469$ $d_4 = 0.0727326195$ $d_5 = -0.0156557281$

2.2.6 Representasi Fitur Citra

Fitur citra dapat direpresentasikan dengan beberapa cara, di antaranya adalah dengan menggunakan nilai *mean* (rata-rata) dan standar deviasi. *Mean* dan standar deviasi dapat digunakan sebagai fitur karena dapat menganalisis suatu tekstur pada domain frekuensi. Perhitungan *mean* dan standar deviasi dari setiap *sub-band* hasil dekomposisi wavelet menghasilkan vektor fitur sehingga dapat disimpan dalam *dataset* fitur citra.

Mean merupakan rata-rata dari suatu sebaran nilai intensitas citra keabuan. *Mean* dirumuskan sebagai berikut [13]:

$$\mu = \frac{1}{M \times N} \sum_{i=1}^M \sum_{j=1}^N X_{ij} \quad (2.3)$$

dengan $M \times N$ adalah ukuran *sub-band* wavelet dan X_{ij} adalah koefisien wavelet.

Sedangkan standar deviasi menunjukkan sebaran nilai piksel pada bidang citra. Standar deviasi dirumuskan sebagai berikut [13]:

$$Std = \left[\frac{1}{M \times N} \sum_{i=1}^M \sum_{j=1}^N (X_{ij} - \mu_{ij})^2 \right]^{\frac{1}{2}} \quad (2.4)$$

dengan $M \times N$ adalah ukuran *sub-band* wavelet, X_{ij} adalah koefisien wavelet dan μ_{ij} adalah nilai rata-rata atau *mean* dari koefisien-koefisien wavelet.

2.2.7 Pengukuran Kemiripan Citra

Tingkat kemiripan atau kesamaan (*similarity*) suatu citra merupakan komponen penting yang mempengaruhi hasil proses suatu algoritma maupun metode. Keakuratan pengukuran kemiripan fitur citra menentukan citra di *database* yang akan ditampilkan sesuai dengan citra *query*.

Perhitungan jarak merupakan salah satu metode yang digunakan untuk mengukur kemiripan fitur suatu citra. Jarak berperan penting dalam metode pengelompokan maupun regresi. Jika jarak antara dua objek semakin besar, maka kemiripan antara kedua objek tersebut semakin rendah.

Beberapa metode perhitungan jarak untuk mengukur tingkat kemiripan suatu citra diantaranya adalah:

1. Jarak Euclidean

Jarak Euclidean merupakan metode perhitungan jarak yang paling umum digunakan. Dua fitur citra yang dibandingkan dan

dihitung dikatakan mirip apabila nilai yang diperoleh adalah nilai terkecil atau mendekati 0. Jika x dan y adalah dua vektor fitur dari citra *query* dan citra *database*, maka jarak Euclidean dari kedua vektor fitur tersebut didefinisikan sebagai berikut [14]:

$$d_i(X, Y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (2.5)$$

2. Jarak Manhattan

Jarak Manhattan disebut juga “*city block distance*” atau blok absolut. Jarak Manhattan mengubah selisih kuadrat dengan menjumlahkan selisih absolut dari variabel-variabel. Sama dengan jarak Euclidean, similaritas atau tingkat kemiripan diperoleh dengan mendapatkan nilai terkecil dari hasil perhitungan. Jarak Manhattan didefinisikan sebagai berikut [14]:

$$d_i(X, Y) = \sum_{i=1}^n |x_i - y_i| \quad (2.6)$$

3. Jarak Canberra

Perhitungan jarak Canberra didasarkan pada nilai mutlak atau absolut dari selisih 2 nilai yang dibagi dengan jumlah dari absolut 2 nilai tersebut. Jika koordinat nol-nol ((0,0)), maka didefinisikan dengan $0/0=0$. Sehingga, jarak Canberra sangat peka terhadap sedikit perubahan dengan kedua koordinat mendekati nol. Jarak Canberra didefinisikan sebagai berikut [14]:

$$d(X, Y) = \sum_{i=1}^n \frac{|x_i - y_i|}{|x_i| + |y_i|} \quad (2.7)$$

BAB III METODOLOGI

Bab ini membahas tentang metode yang digunakan untuk mengerjakan Tugas Akhir. Pembahasan metode yang digunakan berupa penjelasan tentang objek penelitian, peralatan penelitian yang digunakan, tahapan penelitian dan diagram blok.

3.1 Objek Penelitian

Objek penelitian yang digunakan pada Tugas Akhir adalah kain tenun, yaitu meliputi hal-hal sebagai berikut:

1. Jenis Material yang Diteliti
Jenis material yang diteliti pada Tugas Akhir adalah citra tenun yang diambil melalui kamera digital dengan sudut pengambilan gambar yang disesuaikan. Hal ini dilakukan untuk mendapatkan bagian motif yang menjadi ciri khas dari setiap tenun. Citra tenun yang digunakan sebanyak 75 citra.
2. Aspek Penelitian
Aspek penelitian yang digunakan pada Tugas Akhir yaitu mengekstraksi fitur tekstur dari citra tenun dengan memanfaatkan Transformasi Wavelet Diskrit atau *Discrete Wavelet Transform* (DWT) sehingga dapat menemukan citra tenun pada suatu *database* yang memiliki kemiripan dengan citra *query* menggunakan pengukuran kemiripan citra.

3.2 Peralatan Penelitian

Peralatan penelitian yang digunakan untuk mengerjakan Tugas Akhir adalah:

1. Perangkat lunak utama yang digunakan untuk membuat aplikasi Temu Kembali Citra Tenun adalah MATLAB.
2. Personal Computer (PC) untuk menjalankan aplikasi dengan spesifikasi Prosesor Intel(R) Core(TM) i5-5200U CPU @ 2.20GHz 2.20 GHz.

3.3 Tahapan Penelitian

Beberapa tahap yang dilakukan dalam melakukan penelitian pada Tugas Akhir adalah sebagai berikut:

1. Studi Literatur

Salah satu tujuan dilakukan studi literatur adalah untuk membantu mengidentifikasi masalah. Studi literatur pada proses pengerjaan Tugas Akhir dilakukan terhadap beberapa sumber, seperti abstrak hasil penelitian, jurnal dan buku referensi. Studi literatur juga dilakukan untuk mengumpulkan informasi yang terkait dengan penelitian, seperti tentang tenun, *Content Based Image Retrieval* (CBIR), tekstur, Transformasi Wavelet Diskrit dan pengukuran kemiripan citra.

2. Pengumpulan Data

Pengumpulan data dilakukan untuk memperoleh informasi yang dibutuhkan dalam rangka mencapai tujuan Tugas Akhir. Data yang digunakan dalam Tugas Akhir berupa citra tenun dari beberapa daerah yang telah ditentukan.

3. Perancangan dan Implementasi Aplikasi Temu Kembali Citra Tenun

Setelah didapatkan data citra tenun, tahap selanjutnya adalah perancangan aplikasi beserta implementasinya. Aplikasi Temu Kembali Citra Tenun berbasis tekstur dirancang dengan menggunakan Transformasi Wavelet Diskrit. Beberapa tahap utama yang dilakukan berupa baca citra, pra-pemrosesan citra, ekstraksi fitur tekstur dan pengukuran kemiripan citra.

4. Pengujian dan Pembahasan Hasil

Pengujian dilakukan dengan melakukan simulasi aplikasi Temu Kembali Citra Tenun berbasis fitur tekstur terhadap data citra tenun. Selanjutnya, dilakukan pembahasan

terhadap pengujian yang diperoleh. Tahap ini bertujuan untuk membantu memahami cara penggunaan aplikasi sekaligus menganalisis hasil pengujian. Selanjutnya, simpulan dan saran diuraikan berdasarkan analisis hasil pengujian dan pembahasan.

5. Penulisan Laporan Tugas Akhir
Seluruh proses maupun hasil yang didapatkan selama pengerjaan Tugas Akhir ditulis pada tahap ini.

3.4 Diagram Blok

Beberapa tahap utama yang dilakukan pada proses temu kembali citra tenun adalah sebagai berikut:

1. Baca Citra
Baca citra adalah tahap awal untuk pengolahan citra digital. Tahap baca citra dilakukan pada citra tenun, yaitu berupa citra digital yang telah disimpan ke dalam media penyimpanan dan dimasukkan oleh pengguna ke aplikasi. Tahap baca citra dilakukan terhadap citra tenun di *database* dan citra *query*.
2. Pra-Pemrosesan Citra
Pra-pemrosesan citra merupakan tahap yang dilakukan untuk meningkatkan kualitas citra tenun sehingga dapat meningkatkan kemungkinan keberhasilan pada tahap pengolahan citra selanjutnya, yaitu ekstraksi fitur. Rangkaian pra-pemrosesan citra yang dapat dilakukan sebelum tahap ekstraksi fitur adalah *resize* citra dan konversi citra sehingga menjadi citra *grayscale*. Citra yang melewati tahap pra-pemrosesan adalah citra di *database* dan citra *query*.

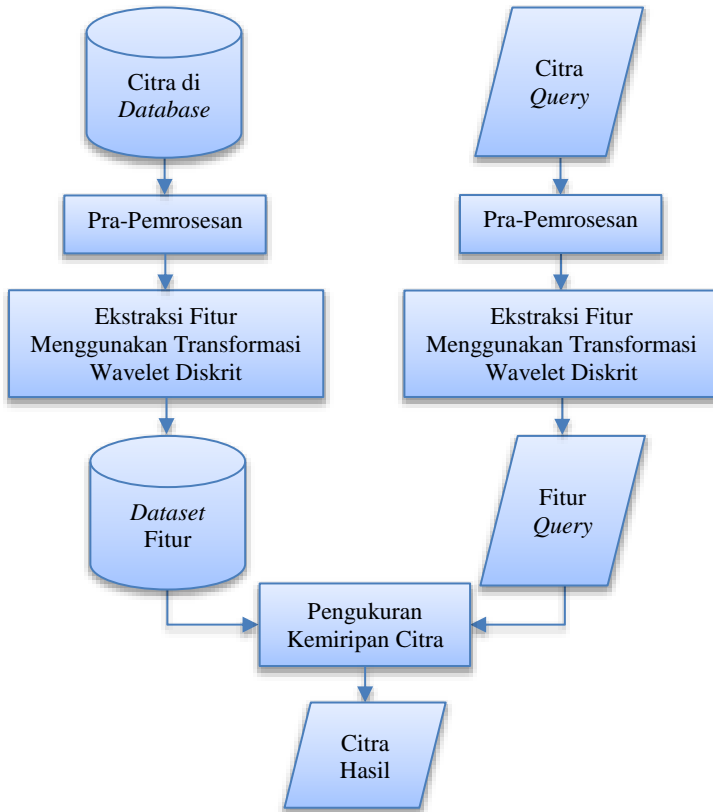
3. Ekstraksi Fitur Tekstur Menggunakan Transformasi Wavelet Diskrit

Tahap ekstraksi fitur tekstur dilakukan pada citra *query* dan citra di *database*. Ekstraksi fitur terdiri dari 2 proses, yaitu proses dekomposisi wavelet terhadap citra dan perhitungan nilai *mean* serta standar deviasi terhadap koefisien-koefisien wavelet hasil proses dekomposisi sebagai representasi dari vektor fitur. Setelah didapatkan *dataset* fitur citra dan fitur *query*, tahap selanjutnya adalah mengukur kemiripan fitur antara citra di *database* dan citra *query*.

4. Pengukuran Kemiripan Citra

Vektor jarak yang dihasilkan dari pengukuran kemiripan citra disimpan secara terurut, yaitu mulai dari jarak yang terkecil sampai jarak yang terbesar. Citra yang ditampilkan dari *database* adalah citra-citra yang memiliki tingkat kemiripan fitur tertinggi dengan citra *query*, yaitu citra yang menempati urutan awal atau citra yang memiliki jarak kecil.

Diagram blok proses temu kembali citra tenun ditunjukkan pada Gambar 3.1.



Gambar 3.1 Diagram Blok Proses Temu Kembali Citra Tenun

BAB IV

PERANCANGAN DAN IMPLEMENTASI

Bab ini menjelaskan tentang perancangan aplikasi beserta implementasi dari semua tahap yang dirancang. Pembahasan perancangan aplikasi diawali dengan penjelasan tentang perancangan data, gambaran aplikasi secara umum, tahap baca citra, tahap pra-pemrosesan citra, tahap ekstraksi fitur tekstur menggunakan Transformasi Wavelet Diskrit dan tahap pengukuran kemiripan citra. Subbab selanjutnya adalah pembahasan tentang implementasi aplikasi yang dimulai dari implementasi antarmuka sampai implementasi tahap-tahap yang telah dirancang sebelumnya.

4.1 Perancangan Aplikasi

Tampilan aplikasi Temu Kembali Citra Tenun dibangun dengan tampilan yang sederhana. Perangkat lunak yang digunakan untuk membangun aplikasi tersebut adalah Matlab.

4.1.1 Perancangan Data

Data yang digunakan pada aplikasi Temu Kembali Citra Tenun dibagi menjadi 3 macam, yaitu data masukan, data proses dan data keluaran. Data masukan adalah data citra tenun hasil pengambilan oleh kamera. Data proses adalah data ketika pemrosesan citra tenun sedang dilakukan. Sedangkan data keluaran berupa data citra hasil proses temu kembali citra tenun setelah dilakukan ekstraksi fitur tekstur menggunakan Transformasi Wavelet Diskrit dan pengukuran kemiripan citra.

1. Data Masukan

Data masukan berupa citra tenun pada bagian motif yang memiliki ciri khas. Citra diambil menggunakan kamera dengan jarak dan sudut yang disesuaikan. Citra tenun terdiri dari 25 citra motif endek asal Bali, 25 citra motif rang-rang asal Jawa Tengah, 25 citra motif lurik asal Nusa Tenggara

Timur, dan 25 motif sikka asal Nusa Tenggara Barat. Sehingga, jumlah citra tenun seluruhnya adalah 100 citra. Citra tenun disimpan dalam format *.jpg*.

2. Data Proses

Data proses merupakan data yang digunakan dalam proses pengolahan data masukan. Data proses diperoleh dari hasil pengolahan data masukan sesuai dengan tahapan metode yang telah disusun. Data proses selengkapnya disajikan pada Tabel 4.1.

Tabel 4.1 Data Proses

Nama Data	Tipe Data	Keterangan
Citra RGB	uint8	Data berupa nilai piksel citra awal.
Citra <i>Grayscale</i>	double	Data ini merupakan citra hasil proses <i>grayscale</i> terhadap citra awal.
Nilai Koefisien Wavelet	double	Data ini merupakan nilai hasil proses dekomposisi wavelet terhadap citra.
Nilai <i>Mean</i> dan Standar Deviasi	double	Data berupa nilai hasil koefisien-koefisien wavelet yang telah diproses.
Vektor Fitur	double	Data ini merupakan matriks yang didapatkan dari nilai standar deviasi.
Vektor Jarak	double	Data ini merupakan hasil pengukuran kemiripan antara <i>dataset</i> vektor fitur dan vektor fitur <i>query</i> .

Tipe data uint8 adalah tipe data yang mempunyai rentang nilai 0-255 (bulat positif). Sedangkan, tipe data double merupakan tipe data primitif yang digunakan untuk

menyimpan bilangan pecahan dan memiliki jangkauan nilai antara $-1,8 \times 10^{308}$ sampai dengan $-5,0 \times 10^{-324}$ dan $5,0 \times 10^{-324}$ sampai dengan $1,8 \times 10^{308}$.

3. Data Keluaran

Data keluaran aplikasi merupakan hasil *retrieval* atau temu kembali citra tenun. Data tersebut berupa sekumpulan citra tenun di *database* yang memiliki tingkat kemiripan fitur tertinggi dengan citra *query*.

4.1.2 Gambaran Aplikasi Secara Umum

Gambaran aplikasi secara umum merupakan gambaran keseluruhan tahap yang dilakukan oleh aplikasi serta algoritma-algoritma yang digunakan untuk mengerjakan masing-masing tahap tersebut. Terdapat 4 tahap utama dalam aplikasi Temu Kembali Citra Tenun, diantaranya adalah sebagai berikut:

1. Tahap Baca Citra Tenun

Tahap baca citra dilakukan pada citra tenun, yaitu berupa citra digital yang telah disimpan ke dalam media penyimpanan dan dimasukkan oleh pengguna ke aplikasi. Tahap baca citra dilakukan terhadap citra tenun di *database* dan citra *query*.

2. Tahap Pra-Pemrosesan

Tahap pra-pemrosesan adalah serangkaian proses untuk mempersiapkan citra tenun sebelum tahap ekstraksi fitur tekstur menggunakan Transformasi Wavelet Diskrit. Tahap pra-pemrosesan citra meliputi beberapa proses, yaitu *resize* citra dan konversi citra sehingga menjadi citra *grayscale*. Citra yang melewati tahap pra-pemrosesan adalah citra di *database* dan citra *query*.

3. Tahap Ekstraksi Fitur Tekstur Menggunakan Transformasi Wavelet Diskrit

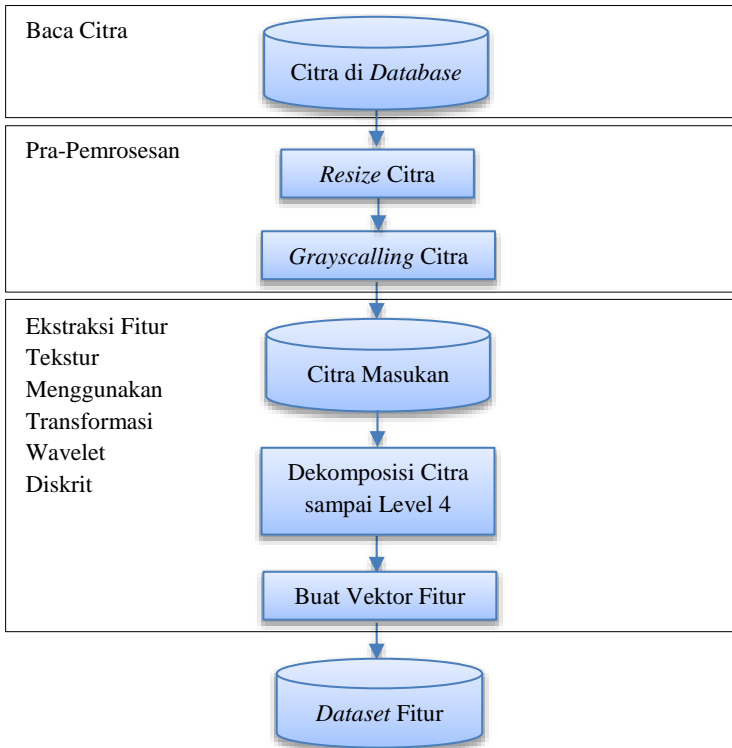
Tahap ekstraksi fitur merupakan serangkaian proses yang dilakukan terhadap citra tenun yang sudah diproses sebelumnya untuk memperoleh fitur tekstur citra tenun sehingga dapat digunakan untuk tahap berikutnya. Tahap ekstraksi fitur dilakukan terhadap citra *query* dan citra di *database*. Hasil ekstraksi fitur terhadap citra di *database* akan disimpan dalam file Matlab dengan format .mat. Tahap ekstraksi fitur terdiri dari 2 proses, yaitu:

- a. Dekomposisi wavelet terhadap citra, adalah proses untuk mendapatkan koefisien-koefisien wavelet menggunakan *Low Pass Filter* (LPF) dan *High Pass Filter* (HPF). Proses tersebut dilakukan sampai level 4.
- b. Perhitungan vektor fitur citra, adalah proses untuk mendapatkan vektor fitur yang direpresentasikan menggunakan nilai *mean* dan standar deviasi. Nilai-nilai tersebut dihitung dari koefisien-koefisien wavelet hasil proses dekomposisi wavelet.

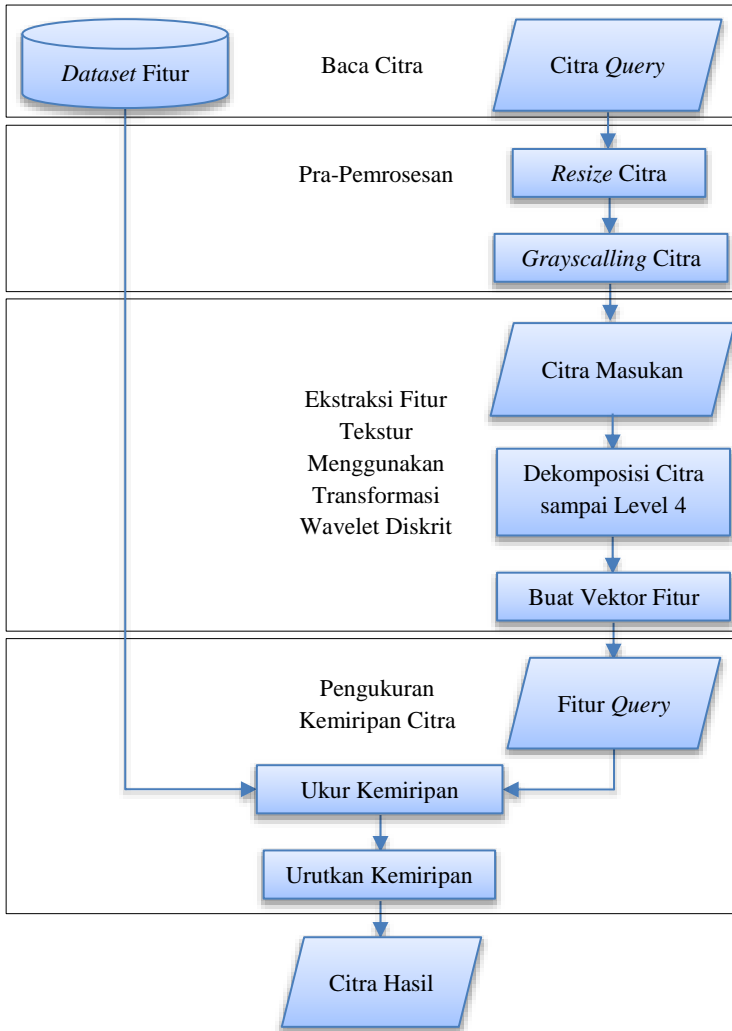
4. Tahap Pengukuran Kemiripan Citra

Tahap pengukuran kemiripan citra merupakan tahap untuk mendapatkan vektor jarak antara vektor fitur citra di *database* dan citra *query* secara terurut, yaitu mulai dari jarak yang terkecil sampai jarak yang terbesar. Citra di *database* yang ditampilkan adalah citra-citra yang memiliki tingkat kemiripan fitur tertinggi dengan citra *query*, yaitu citra-citra yang menempati urutan awal atau citra-citra yang memiliki jarak kecil.

Gambaran rangkaian proses aplikasi Temu Kembali Citra Tenun menggunakan Transformasi Wavelet Diskrit dapat dilihat pada Gambar 4.1 dan Gambar 4.2.



Gambar 4.1 Diagram Alir Proses Temu Kembali pada Citra Tenun di *Database*



Gambar 4.2 Diagram Alir Proses Temu Kembali Citra Tenun

4.1.3 Tahap Baca Citra Tenun

Baca citra tenun merupakan tahap awal yang dilakukan terhadap citra yang telah diakuisisi oleh pengguna. Akuisisi citra tenun merupakan proses pengambilan citra tenun menggunakan kamera digital sehingga dapat dilakukan serangkaian proses terhadap citra tersebut. Pengambilan gambar dilakukan dengan melakukan pengaturan posisi kain tenun dan kamera agar menghasilkan lingkungan yang sesuai. Lingkungan yang sesuai meliputi pencahayaan yang cukup, jarak antara kamera dan kain tenun dan bagian kain yang menonjolkan ciri khas motif. Hal ini bertujuan agar diperoleh citra tenun yang lebih mudah diproses nantinya. Citra tenun tersebut disimpan dalam format *.jpg*.

Data citra tenun yang diperoleh adalah sebanyak 100 citra yang terdiri dari 25 citra motif endek asal Bali, 25 citra motif rangrang asal Jawa Tengah, 25 citra motif lurik asal Nusa Tenggara Timur, dan 25 motif sikka asal Nusa Tenggara Barat. Citra tersebut digunakan dalam tahap baca citra dari rangkaian proses temu kembali citra tenun.

4.1.4 Tahap Pra-Pemrosesan Citra Tenun

Sebelum memperoleh fitur suatu citra, terlebih dahulu dilakukan pra-pemrosesan untuk memastikan data yang akan diekstraksi merupakan data yang baik. Proses-proses tersebut diantaranya adalah sebagai berikut:

1. Proses *resize*, yaitu proses untuk mengubah ukuran panjang dan lebar citra yang telah diakuisisi. Citra tenun awal dilakukan proses *resize* sehingga direpresentasikan ke dalam matriks berukuran 128x128 piksel.
2. Proses *grayscale*, yaitu proses mengubah citra berwarna (RGB) menjadi citra *gray* atau citra yang memiliki nilai keabuan. Proses *grayscale* mengubah citra RGB yang memiliki tiga *channel* warna, yaitu *channel* warna merah, hijau dan biru menjadi citra *grayscale*. Setelah citra tenun melalui proses *grayscale*, dapat dilanjutkan ke tahap ekstraksi fitur tekstur.

4.1.5 Tahap Ekstraksi Fitur Tekstur Menggunakan Tranformasi Wavelet Diskrit

Ekstraksi fitur pada citra tenun bertujuan untuk menentukan fitur atau ciri suatu citra tenun sehingga dapat dibedakan antara citra tenun yang satu dengan yang lainnya. Fitur yang diekstraksi dari citra tenun adalah fitur tekstur. Metode Transformasi Wavelet Diskrit merupakan salah satu metode yang dapat digunakan untuk mengekstraksi fitur tekstur suatu citra, khususnya citra tenun. Tahap ini merupakan pemilihan jenis wavelet yang digunakan, yaitu Wavelet Haar, Wavelet Daubechies 4, Wavelet Symmlet 2 atau Wavelet Coiflet 1. Proses-proses yang dilakukan untuk mengekstraksi citra adalah sebagai berikut:

1. Mendekomposisi citra tenun hasil pra-pemrosesan sampai level 4.
2. Menghitung vektor fitur yang direpresentasikan dengan nilai *mean* dan standar deviasi dari nilai-nilai koefisien wavelet hasil proses dekomposisi.

Berikut penjelasan dari kedua proses yang dilakukan untuk mengekstraksi citra.

1. Proses Dekomposisi

Algoritma dekomposisi Transformasi Wavelet Diskrit adalah membagi atau mendekomposisi sebuah dimensi sinyal menjadi dua bagian, disebut bagian dengan frekuensi tinggi dan frekuensi rendah. Sebuah sinyal dilewatkan melalui *High Pass Filter* (HPF) dan *Low Pass Filter* (LPF). Selanjutnya dilakukan proses *sub-band*, yaitu mengambil *sample* dari setengah keluaran pada masing-masing *filter* tersebut. Sehingga, proses dekomposisi level 1 menghasilkan 4 buah *sub-band*. Tugas Akhir ini melakukan proses dekomposisi sampai level 4. Untuk melakukan proses dekomposisi lebih dari satu kali, proses dekomposisi selanjutnya dilakukan pada *sub-band* frekuensi rendah – frekuensi rendah atau LL. Sehingga, dengan dekomposisi

level 4 didapatkan 16 *sub-band*. Selanjutnya, nilai standar deviasi dari setiap *sub-band* dapat dihitung.

2. Proses Perhitungan Vektor Fitur

Vektor fitur citra hasil dekomposisi dengan wavelet diperoleh dengan menghitung *mean* dan standar deviasi yang terdapat pada setiap *sub-band*. Untuk membuat *dataset* vektor fitur citra, proses ini dilakukan secara berulang terhadap semua citra yang berada dalam *database*. Nilai *mean* dan standar deviasi inilah yang kemudian disimpan ke dalam *dataset* vektor fitur sebagai representasi fitur tekstur dari setiap citra.

4.1.6 Tahap Pengukuran Kemiripan Citra Tenun

Tahap pengukuran kemiripan citra antara vektor fitur citra *query* dengan vektor fitur citra di *database* akan menghasilkan rentang jarak kemiripan pada setiap *sub-band* citra *query* dengan citra di *database*. Semakin kecil rentang jarak, maka kedua citra semakin mirip. Rentang jarak yang berupa vektor tersebut disimpan secara terurut dimulai dari jarak yang terkecil. Citra yang ditampilkan adalah citra yang memiliki kemiripan tekstur dengan citra *query*, yaitu citra dengan jarak yang kecil atau mendekati 0.

Tugas Akhir ini menggunakan 3 metode perhitungan jarak, yaitu jarak Euclidean, jarak Manhattan dan jarak Canberra. Ketiga metode perhitungan jarak digunakan sebagai perbandingan.

Selanjutnya, terdapat 2 pilihan cara untuk menampilkan citra di *database* yang memiliki tingkat kemiripan fitur tertinggi dengan citra *query*, diantaranya adalah sebagai berikut:

1. Mengatur jumlah citra yang ingin ditampilkan, mulai dari 1 citra sampai 15 citra.
2. Mengatur nilai ambang batas atau *threshold* untuk membatasi jumlah citra yang ditampilkan.

4.2 Implementasi

Perancangan aplikasi yang telah dibangun, selanjutnya diimplementasikan pada bahasa pemrograman menggunakan *software* Matlab. Pembahasan dalam implementasi aplikasi meliputi implementasi antarmuka (*interface*) aplikasi, implementasi tahap baca citra, implementasi tahap pra-pemrosesan citra, implementasi tahap ekstraksi fitur tekstur menggunakan Transformasi Wavelet Diskrit, dan tahap pengukuran kemiripan citra dengan metode pengukuran kemiripan.

4.2.1 Implementasi Antarmuka

Antarmuka aplikasi pada Tugas Akhir dibangun dengan menggunakan *form* dan kontrol yang terdapat pada Matlab. Adapun antarmuka-antarmuka yang diimplementasikan untuk menunjang penelitian Tugas Akhir adalah sebagai berikut:

1. Antarmuka Utama

Antarmuka Utama merupakan antarmuka yang berisi menu-menu untuk menampilkan antarmuka-antarmuka lainnya dalam aplikasi. Hasil implementasi antarmuka Utama ditunjukkan pada Gambar 4.3.



Gambar 4.3 Antarmuka Utama

Antarmuka Utama dibuat dalam bentuk sederhana yang terdiri dari 3 bagian, yaitu *title bar*, *menu bar* dan *main window*. *Title bar* merupakan bagian yang menunjukkan judul antarmuka yang sedang ditampilkan. Di bawah *title bar* terdapat *menu bar* yang berisi sederet menu yang digunakan dalam aplikasi. Sedangkan *Main window* merupakan bagian antarmuka yang digunakan untuk menampilkan berbagai antarmuka lain di dalam aplikasi. Kegunaan menu-menu yang ditampilkan pada halaman utama pada aplikasi disajikan dalam Tabel 4.2.

Tabel 4.2 Kegunaan Menu Aplikasi

Menu	Kegunaan
Halaman Utama	Mengembalikan tampilan aplikasi ke halaman utama.
Proses dan Uji	Melakukan proses terhadap data masukan yang kemudian semua data disimpan dalam aplikasi dan melakukan pengujian, pengamatan, serta penilaian terhadap aplikasi, seberapa baik hasil yang didapatkan pada data masukan tertentu.
Keluar	Keluar dari aplikasi.

2. Antarmuka Proses dan Uji

Antarmuka Proses dan Uji berguna untuk melakukan proses terhadap data masukan yang kemudian semua data disimpan dalam aplikasi dan melakukan pengujian, pengamatan, serta penilaian terhadap aplikasi, seberapa baik hasil yang didapatkan pada data masukan tertentu. Seperti antarmuka Utama, antarmuka Proses dan Uji juga memiliki menu file yang dapat digunakan untuk mempermudah proses uji coba. Hasil implementasi antarmuka Proses dan Uji ditunjukkan pada Gambar 4.4.



Gambar 4.4 Antarmuka Proses dan Uji

Antarmuka Proses dan Uji memiliki 6 *panel* yang masing-masing berisi beberapa *push button*, *pop-up menu*, *edit text* atau *axes*. Komponen-komponen tersebut memiliki fungsi yang berbeda-beda sebagai berikut:

1. *Panel* ‘Input Dataset’
 - a. *Pop-up menu* ‘Pilih Wavelet’, terdiri dari 4 pilihan menu yaitu *Wavelet Haar*, *Wavelet Daubechies*, *Wavelet Symlets* dan *Wavelet Coiflets*. Masing-masing dari pilihan menu tersebut digunakan untuk menentukan jenis wavelet yang akan digunakan ketika ekstraksi data citra dilakukan.
 - b. *Push Button* ‘Pilih Direktori Citra’, digunakan untuk memilih folder citra tenun yang tersimpan pada komputer. Setelah folder citra tersebut dipilih, maka dapat diproses dan disimpan sebagai *dataset*.
 - c. *Push Button* ‘Buat Dataset’, digunakan untuk melakukan pemrosesan terhadap data citra pada folder yang telah dipilih. Pemrosesan tersebut meliputi proses *resize*, proses *grayscale* dan proses ekstraksi citra. Proses ekstraksi dilakukan sesuai dengan jenis wavelet yang

telah dipilih sebelumnya. Apabila *push button* tersebut dipilih, maka akan dilakukan pemrosesan selama beberapa saat. Hasil dari pemrosesan tersebut berupa *dataset* fitur citra yang dapat disimpan pada aplikasi dalam format .mat oleh pengguna.

- d. *Push Button* ‘Load Dataset’, digunakan untuk memuat atau mengambil *dataset* fitur citra tenun yang telah tersimpan pada aplikasi.

2. *Panel* ‘Input Query’

- a. *Pop-up menu* ‘Pilih Wavelet’, terdiri dari 4 pilihan menu yaitu *Wavelet* Haar, *Wavelet* Daubechies, *Wavelet* Symlets dan *Wavelet* Coiflets. Seperti pada *panel* ‘Input Dataset’, masing-masing dari pilihan menu tersebut digunakan untuk menentukan jenis wavelet yang akan digunakan ketika ekstraksi citra dilakukan, namun hanya untuk citra *query*.
- b. *Push button* ‘Pilih Citra Query’, digunakan untuk memasukkan citra *query* yang tersimpan pada komputer dan melakukan pemrosesan terhadap citra tersebut. Pemrosesan yang dilakukan sama seperti pemrosesan ketika membuat *dataset*, yaitu meliputi proses *resize*, proses *grayscale* dan proses ekstraksi citra. Proses ekstraksi dilakukan sesuai dengan jenis wavelet yang telah dipilih sebelumnya. Hasil dari pemrosesan tersebut berupa fitur citra *query* yang langsung tersimpan pada aplikasi.

3. *Panel* ‘Ukuran Kemiripan’

Terdapat 3 pilihan metode perhitungan jarak yang dapat digunakan untuk melakukan proses pengukuran kemiripan citra terhadap *dataset* fitur dan fitur *query* yang telah tersimpan. Pilihan metode tersebut diantaranya adalah jarak Euclidean, jarak Manhattan dan jarak Canberra.

4. *Panel ‘Atur Jumlah Citra Kembali’*
 - a. *Pop-up menu ‘Jumlah Citra’*, terdapat 15 angka pilihan yaitu angka 1 sampai 15. Pilihan angka tersebut digunakan untuk menentukan jumlah citra yang akan ditampilkan pada *axes* sebagai citra dari *database* yang memiliki kemiripan fitur tertinggi dengan citra *query*.
 - b. *Push button ‘Temukan Citra’* warna coklat, digunakan untuk melakukan pemrosesan terhadap *dataset* citra tenun dan citra *query* yang telah dipilih sebelumnya dengan menggunakan metode wavelet dan metode pengukuran kemiripan yang juga telah dipilih, sehingga dapat ditemukan citra di *database* yang memiliki kemiripan fitur tertinggi dengan citra *query*. Hasil pencarian citra tersebut akan ditampilkan sesuai dengan nilai jumlah citra dicari yang ditentukan oleh pengguna.

5. *Panel ‘Atur Threshold’*
 - a. *Edit text ‘Threshold’*, digunakan untuk memasukkan nilai *threshold* atau nilai ambang batas tertentu pada aplikasi. Nilai *threshold* berpengaruh pada jumlah citra yang ditampilkan pada *axes*, yaitu hanya citra dari *database* yang memiliki kemiripan fitur tertinggi dengan citra *query* dan memiliki nilai vektor jarak atau nilai kemiripan yang tidak melebihi nilai *threshold*.
 - b. *Push button ‘Temukan Citra’* warna hijau, sama seperti ‘Temukan Citra’ warna coklat, digunakan untuk melakukan pemrosesan terhadap *dataset* fitur citra tenun dan citra *query* yang telah dipilih sebelumnya dengan menggunakan metode wavelet dan metode pengukuran kemiripan yang juga telah dipilih, sehingga dapat ditemukan citra di *database* yang memiliki kemiripan fitur tertinggi dengan citra *query*. Namun, hasil pencarian citra yang akan ditampilkan bergantung pada nilai *threshold* atau nilai ambang batas yang ditentukan oleh pengguna.

6. *Axes* ‘Hasil Temu Kembali Citra Tenun’

Sebagai tempat untuk menampilkan citra *query* beserta hasil keseluruhan proses, yaitu citra-citra yang ditampilkan sebagai citra di *database* yang mirip dengan citra *query*.

4.2.2 Implementasi Tahap Baca Citra Tenun

Tahap baca citra digunakan untuk mendapatkan citra tenun yang telah tersimpan sebelumnya. Tahap ini tidak dilakukan secara otomatis, tapi membutuhkan interaksi pengguna untuk mencari dan mengambil citra yang dibutuhkan dari media penyimpanan.

Terdapat 2 tahap baca citra, yaitu tahap baca citra pada folder citra untuk membuat *dataset* dan tahap baca citra untuk citra *query*. Kedua tahap tersebut masing-masing diimplementasikan menjadi sebuah program ke dalam fungsi-fungsi berikut:

```
image = imread(fullfile(handles.folder_name,
jpg_files(jpg_counter+1).name));
```

```
queryImage =
imread(fullfile(pathstr, strcat(name, ext)));
```

Masing-masing fungsi di atas dijalankan ketika salah satu *push button* jenis wavelet dipilih.

4.2.3 Implementasi Tahap Pra-Pemrosesan Citra Tenun

Tahap pra-pemrosesan pada Tugas Akhir terdiri dari beberapa proses yang akan mempengaruhi tahap selanjutnya, begitu juga dengan hasil yang akan didapatkan. Proses-proses tersebut diantaranya adalah proses *resize* dan *grayscale*.

1. Implementasi Proses *Resize*

Proses *resize* bertujuan mengubah ukuran citra menjadi citra normal sesuai dengan yang diinginkan sehingga setiap citra akan

berukuran sama sebelum dilakukan proses selanjutnya. Proses *resize* terhadap citra menggunakan fungsi pada matlab sebagai berikut :

```
image = imresize(image, [128 128]);
```

Hasil dari proses *resize* adalah data keluaran berukuran 128x128 piksel.

2. Implementasi Proses *Grayscale*

Proses *grayscale* bertujuan untuk mengubah citra berwarna menjadi citra dalam bentuk *grayscale*. Proses tersebut juga dapat menggunakan fungsi pada matlab dengan fungsi `rgb2gray`.

```
imgGray = double(rgb2gray(image))/255;
```

Data keluaran fungsi tersebut adalah citra tenun dalam bentuk *grayscale* atau citra tenun yang memiliki derajat keabuan.

4.2.4 Implementasi Tahap Ekstraksi Fitur Tekstur Menggunakan Transformasi Wavelet Diskrit

Setelah dilakukan pra-pemrosesan, tahap selanjutnya adalah ekstraksi fitur tekstur. Tahap ekstraksi bertujuan untuk mendapatkan vektor fitur dari citra tenun sehingga dapat dilakukan tahap pengukuran kemiripan citra dengan baik.

Citra *grayscale* hasil pra-pemrosesan digunakan sebagai *input* atau masukan dalam tahap ekstraksi fitur tekstur dengan Transformasi Wavelet Diskrit sampai level 4. Terdapat 4 pilihan jenis wavelet yang digunakan, yaitu Wavelet Haar, Wavelet Daubechies 4, Wavelet Symmlet 2 dan Wavelet Coiflet 1. Masing-masing jenis wavelet untuk ekstraksi tersebut diimplementasikan menjadi sebuah program dalam fungsi-fungsi berikut:

```
function waveletMoments =  
waveletHaarTransform(image)
```

```
function waveletMoments =  
waveletDaubechiesTransform(image)
```

```
function waveletMoments =  
waveletSymletsTransform(image)
```

```
function waveletMoments =  
waveletCoifletsTransform(image)
```

Keluaran fungsi-fungsi tersebut adalah vektor fitur yang merupakan nilai *mean* dan standar deviasi dari koefisien wavelet. Kode selengkapnya dari masing-masing fungsi tersebut disajikan pada Lampiran A1, A2, A3 dan A4.

4.2.5 Implementasi Tahap Pengukuran Kemiripan Citra Tenun

Tahap selanjutnya adalah pengukuran kemiripan terhadap vektor fitur hasil tahap ekstraksi. Jarak antara setiap fitur citra di *dataset* dan fitur citra *query* dihitung menggunakan salah satu metode pengukuran kemiripan yang disediakan, yaitu jarak Euclidean, jarak Manhattan dan Jarak Canberra. Implementasi tahap pengukuran kemiripan tersebut berupa fungsi-fungsi sebagai berikut:


```
function Euclidean(numOfReturnedImages,
queryImageFeatureVector, dataset)
```

```
function Manhattan(numOfReturnedImages,
queryImageFeatureVector, dataset)
```

```
function Canberra(numOfReturnedImages,
queryImageFeatureVector, dataset)
```

Fungsi-fungsi tersebut menghasilkan nilai kemiripan yang belum terurut. Selanjutnya, nilai kemiripan yang telah didapatkan harus melalui proses pengurutan dari nilai terkecil sampai nilai terbesar agar bisa menampilkan citra tenun dari *database* yang termirip dengan citra tenun *query*. Proses tersebut diimplementasikan pada program ke dalam fungsi berikut:

```
[sortedDist indxs] = sortrows(A)
```

Variabel A merupakan variabel yang berisi vektor kemiripan yang belum terurut. Selanjutnya, untuk menampilkan citra tenun termirip yaitu fitur citra pada *dataset* yang memiliki tingkat kemiripan fitur tertinggi dengan mengambil urutan pertama sampai urutan ke-15 atau dengan menggunakan nilai *threshold*. Nilai *threshold* merupakan nilai ambang batas yang membatasi jumlah citra termirip yang ditampilkan. Kode selengkapnya disajikan pada Lampiran B1, B2 dan B3 untuk proses pengukuran kemiripan dengan mengatur jumlah citra kembali. Sedangkan Lampiran B4, B5 dan B6 untuk proses pengukuran kemiripan dengan mengatur nilai *threshold*.

BAB V

PENGUJIAN DAN PEMBAHASAN HASIL

Bab ini menjelaskan tentang proses pengujian yang dilakukan terhadap aplikasi Temu Kembali Citra Tenun menggunakan Transformasi Wavelet Diskrit beserta pembahasannya. Pengujian terhadap aplikasi dilakukan untuk mengetahui unjuk kerja aplikasi secara keseluruhan dalam menjalankan fungsi-fungsi yang diinginkan. Bab ini diakhiri dengan pembahasan hasil pengujian yang telah dilakukan.

5.1 Pengujian Aplikasi

Pengujian aplikasi dimulai dengan pengenalan lingkungan pengujian aplikasi yang digunakan. Selanjutnya, dijelaskan mengenai hasil pengujian terhadap aplikasi yang telah diimplementasikan pada bab sebelumnya, yaitu pengujian tahap baca citra, tahap pra-pemrosesan, tahap ekstraksi fitur tekstur menggunakan Transformasi Wavelet Diskrit dan pengukuran kemiripan citra dengan beberapa parameter.

5.1.1 Lingkungan Pengujian Aplikasi

Lingkungan pengujian aplikasi Temu Kembali Citra Tenun meliputi perangkat keras dan perangkat lunak komputer. Detail perangkat keras dan perangkat lunak yang digunakan dapat dilihat pada Tabel 5.1.

Tabel 5.1 Lingkungan Pengujian Aplikasi

Jenis Perangkat	Detail Perangkat
Perangkat Keras	Prosesor : Intel® Core(TM) i5-5200U CPU @ 2.20GHz 2.20 GHz
	Memory : 4 GB DDR3
Perangkat Lunak	Sistem Operasi : Windows 10 Pro 64-bit
	Tools : MATLAB

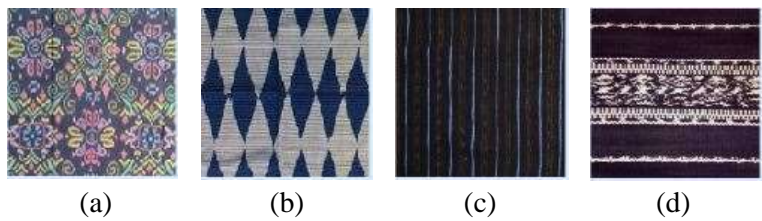
5.1.2 Pengujian Tahap Baca Citra Tenun

Pengujian tahap baca citra dilakukan terhadap data citra tenun yang akan digunakan sebagai bahan dasar untuk proses citra selanjutnya. Data yang digunakan secara keseluruhan berjumlah 100 citra. Data tersebut dibagi menjadi 2 jenis, yaitu 80% dari citra keseluruhan atau sebanyak 80 citra sebagai citra di *database* dan 20% sisanya atau sebanyak 20 citra sebagai *query*. Masing-masing citra berukuran 300x300 piksel. Kedua jenis data citra tersebut terdiri dari 4 kelompok dengan setiap kelompok terdiri dari sejumlah citra sejenis yang selanjutnya dikatakan relevan. Detail data citra yang digunakan disajikan pada Tabel 5.2.

Tabel 5.2 Kelompok Citra Tenun

Kelompok	Nama Semantik	<i>Database</i>	<i>Query</i>
1	Tenun Bali	20	5
2	Tenun Jawa Tengah	20	5
3	Tenun NTT	20	5
4	Tenun NTB	20	5

Gambar 5.1 menunjukkan gambar hasil tahap baca citra tenun dari masing-masing kelompok.



Gambar 5.1 Hasil Baca Citra Tenun.

(a) Citra Tenun Bali; (b) Citra Tenun Jawa Tengah;
(c) Citra Tenun NTT; (d) Citra Tenun NTB.

5.1.3 Pengujian Tahap Pra-Pemrosesan Citra Tenun

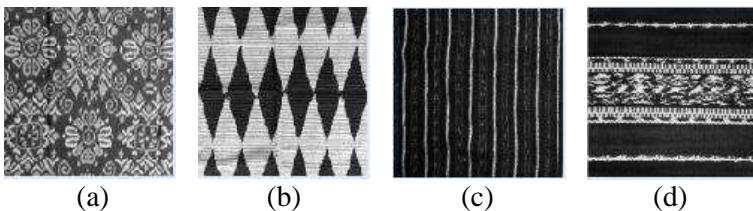
Pengujian tahap pra-pemrosesan bertujuan untuk mengetahui bahwa pra-pemrosesan terhadap citra sudah benar sehingga data yang dihasilkan dipastikan dapat menjadi data masukan pada tahap berikutnya. Tahap pra-pemrosesan terdiri dari proses *resize* citra dan proses *grayscale* citra. Hasil pra-pemrosesan ditunjukkan pada Gambar 5.2.

1. Pengujian proses *resize*

Pengujian proses *resize* bertujuan untuk mengetahui bahwa aplikasi telah mengubah ukuran piksel citra tenun menjadi 128x128.

2. Pengujian *grayscale*

Pengujian *grayscale* bertujuan untuk mengetahui bahwa aplikasi telah mengubah citra tenun hasil proses *resize* yang masih dalam bentuk citra berwarna menjadi citra *grayscale*.



Gambar 5.2 Hasil Pra-Pemrosesan *Resize* dan *Grayscale*.


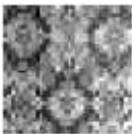
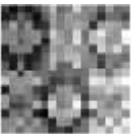
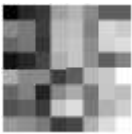
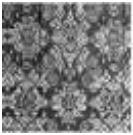


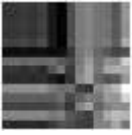

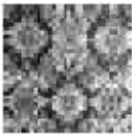
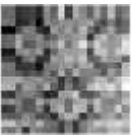





(a) Citra Tenun Bali; (b) Citra Tenun Jawa Tengah;
(c) Citra Tenun NTT; (d) Citra Tenun NTB.

5.1.4 Pengujian Tahap Ekstraksi Fitur Tekstur Menggunakan Transformasi Wavelet Diskrit

Pengujian tahap ekstraksi fitur tekstur menggunakan data citra tenun yang telah melalui tahap pra-pemrosesan. Data masukan yang telah diproses sebelumnya berupa citra *grayscale* tenun. Ekstraksi fitur tekstur dilakukan terhadap seluruh data citra

grayscale, yaitu sejumlah 100 citra. Data masukan tersebut didekomposisi sampai level 4 menggunakan 4 wavelet yang berbeda, yaitu Wavelet Haar, Wavelet Daubechies 4, Wavelet Symlets 2 dan Wavelet Coiflets 1. Hasil dekomposisi terhadap Gambar 5.2 (a) sebagai citra *query* disajikan pada Tabel 5.3.

Tabel 5.3 Hasil Dekomposisi Wavelet Level 1 sampai Level 4

Jenis Wavelet	Level 1	Level 2	Level 3	Level 4
Haar				
Daub 4				
Sym 2				
Coif 1				

Selanjutnya, hasil dekomposisi yang berupa koefisien wavelet digunakan untuk menghitung nilai *mean* dan standar

deviasi sebagai representasi dari fitur citra. Sehingga, tahap ekstraksi fitur tekstur menggunakan Transformasi Wavelet Diskrit menghasilkan 4 data keluaran yaitu fitur yang berupa vektor kolom dengan ukuran berbeda. Ukuran vektor kolom yang dihasilkan dari ekstraksi terhadap citra di *database* dengan masing-masing wavelet yang digunakan disajikan pada Tabel 5.3.

Tabel 5.4 Hasil Ekstraksi Fitur Tekstur Citra di *Database* Menggunakan Transformasi Wavelet Diskrit

Jenis Wavelet	Value	Waktu Komputasi
Haar	80 x 17 <i>double</i>	2,87 detik
Daub 4	80 x 29 <i>double</i>	3,02 detik
Sym 2	80 x 21 <i>double</i>	2,90 detik
Coif 1	80 x 25 <i>double</i>	2,95 detik

Data keluaran dari tahap ekstraksi fitur tekstur terhadap citra di *database* disimpan dalam file Matlab dengan format *.mat*. Sedangkan ukuran vektor kolom yang dihasilkan dari tahap ekstraksi terhadap setiap citra *query* dengan masing-masing wavelet yang digunakan secara terurut seperti pada Tabel 5.3 adalah 1 x 17 *double*, 1 x 29 *double*, 1 x 21 *double*, 1 x 25 *double*. Kedua hasil tahap ekstraksi yang berupa fitur tersebut digunakan sebagai data masukan dalam tahap pengukuran kemiripan citra.

5.1.5 Pengujian Tahap Pengukuran Kemiripan Citra Tenun

Pengujian tahap pengukuran kemiripan citra tenun dilakukan dengan membandingkan nilai-nilai kemiripan yang berupa jarak antara fitur citra *dataset* dengan fitur citra *query*. Nilai jarak yang terkecil menunjukkan kemiripan tertinggi antara citra di *database* dengan citra *query*.

Pengujian dilakukan terhadap 5 citra tenun dari setiap kelompok atau dari setiap daerah. Sehingga, citra *query* yang diuji adalah sebanyak 20 citra tenun. Setiap citra *query* dibandingkan dengan 80 citra tenun di *database* untuk setiap jenis wavelet.

Pengujian tahap pengukuran kemiripan citra dilakukan dengan menggunakan metode perhitungan jarak, diantaranya adalah jarak Euclidean, jarak Manhattan dan jarak Canberra.

Selanjutnya, terdapat 2 pilihan cara pengujian untuk menampilkan citra di *database* yang memiliki tingkat kemiripan fitur tertinggi dengan citra *query*, diantaranya adalah sebagai berikut:

- 1. Pengujian dengan mengatur jumlah citra yang ingin ditampilkan

Tabel 5.5 Hasil Pengukuran Kemiripan Citra dengan Mengatur Jumlah yang Ingin Ditampilkan

Citra Query	Hasil Temu Kembali		
			
			

Tabel 5.5 menunjukkan kumpulan citra tenun yang didapatkan sebagai hasil tahap pengukuran kemiripan fitur citra dengan mengatur jumlah citra hasil sebanyak 6 citra. Pengujian dilakukan menggunakan Wavelet Haar dan Jarak Euclidean.






Data keluaran dari tahap pengukuran kemiripan fitur citra berupa nilai jarak antara fitur *query* dengan *dataset* fitur yang telah terurut dari nilai terkecil sampai terbesar. Citra-citra yang memiliki nilai jarak terkecil akan ditampilkan sebagai citra yang memiliki

kemiripan tertinggi dengan citra *query*. Seperti pada Tabel 5.5, ditampilkan 6 citra yang memiliki kemiripan tertinggi dengan citra *query*. Namun, diantara 6 citra yang telah ditampilkan terdapat 2 citra yang tidak relevan dengan citra *query*, yaitu citra ke-5 dan citra ke-6.

2. Pengujian dengan mengatur nilai ambang batas atau *threshold*

Untuk mengatasi permasalahan pada pengujian dengan mengatur jumlah citra yang ingin ditampilkan, dibutuhkan nilai *threshold* atau nilai ambang batas sehingga dapat memperkecil nilai *error*. Nilai *threshold* dibandingkan dengan nilai jarak mulai dari nilai yang terkecil. Apabila nilai jarak lebih kecil dari nilai *threshold*, maka citra relevan dengan citra *query*. Namun, apabila nilai jarak lebih besar dari nilai *threshold*, maka menunjukkan bahwa citra tidak relevan dengan citra *query*.

Tabel 5.6 Hasil Pengukuran Kemiripan Citra dengan Mengatur Nilai *Threshold*

Citra Query	Hasil Temu Kembali		
			
			

Hasil tahap pengukuran kemiripan fitur citra untuk citra *query* yang sama dengan pengujian sebelumnya, dengan nilai

threshold 2,7 ditunjukkan pada Tabel 5.6. Pengujian dilakukan menggunakan Wavelet Haar dan Jarak Euclidean.

Selanjutnya, pengujian dilakukan dengan mengubah-ubah nilai *threshold* yang digunakan. Hal ini bertujuan untuk menentukan nilai *threshold* terbaik sehingga tahap pengukuran kemiripan citra dapat mencapai hasil optimal. Nilai-nilai *threshold* tersebut didapatkan dari hasil pengukuran kemiripan antara masing-masing citra di *database*. Pengujian dilakukan terhadap 20 citra *query* untuk setiap metode perhitungan jarak.

Perhitungan persentase nilai presisi dilakukan untuk mengukur ketepatan dari hasil pengujian yang didapatkan. Persentase nilai Presisi (P) dari satu citra *query* dirumuskan sebagai berikut [15]:

$$P(k) = \frac{n_k}{L} \times 100 \quad (5.1)$$

dengan n_k adalah jumlah citra yang relevan pada citra *query* ke- k yang ditampilkan dan L adalah jumlah citra yang ditampilkan.

Sedangkan rata-rata persentase nilai presisi (μ_P) dirumuskan sebagai berikut:

$$\mu_P = \frac{\sum_{i=1}^m P_i}{m} \quad (5.2)$$

dengan m adalah jumlah citra *query* yang diuji.

Hasil pengujian untuk masing-masing metode jarak, yaitu jarak Euclidean, jarak Manhattan dan jarak Canberra dengan nilai *threshold* dan jenis wavelet yang berbeda disajikan pada tabel-tabel berikut:

1. Pengujian dengan jarak Euclidean

Hasil pengukuran kemiripan citra menggunakan jarak Euclidean dengan nilai *threshold* dan jenis wavelet yang berbeda ditunjukkan pada Tabel 5.7.

Tabel 5.7 Hasil Pengukuran Kemiripan Citra Menggunakan Jarak Euclidean

Jenis Wavelet	Nilai <i>Threshold</i>	Citra Relevan	Citra Tidak Relevan	Rata-Rata Persentase Presisi
Haar	3,0	111	33	83,59%
	3,1	114	40	81,27%
	3,2	120	48	78,55%
	3,3	127	61	74,28%
Daub 4	5,8	114	117	58,24%
	5,9	116	122	57,40%
	6,0	117	123	57,29%
	6,1	118	125	55,85%
Sym 2	5,1	143	91	66,27%
	5,2	144	92	65,82%
	5,3	146	93	65,77%
	5,4	147	100	64,22%
Coif 1	5,2	114	93	64,32%
	5,3	116	94	63,43%
	5,4	124	96	64,19%
	5,5	126	98	63,94%

Hasil pengukuran dengan jarak Euclidean menunjukkan rata-rata persentase presisi terbesar untuk Wavelet Haar adalah 83,59% dengan nilai *threshold* 3,0, untuk Wavelet Daubechies 4 adalah 58,24% dengan nilai *threshold* 5,8, untuk Wavelet Symlets 2 adalah 66,27% dengan nilai *threshold* 5,1 dan untuk Wavelet Coiflets 1 adalah 64,32% dengan nilai *threshold* 5,2.

Hasil perbandingan presisi menunjukkan bahwa Wavelet Haar memperoleh rata-rata persentase presisi terbesar, yaitu 83,59%. Sedangkan Wavelet Daubechies 4 memperoleh rata-rata persentase presisi terkecil, yaitu 58,24%.

2. Pengujian dengan jarak Manhattan

Hasil pengukuran kemiripan citra menggunakan jarak Manhattan dengan nilai *threshold* dan jenis wavelet yang berbeda-beda ditunjukkan pada Tabel 5.8.

Tabel 5.8 Hasil Pengukuran Kemiripan Citra Menggunakan Jarak Manhattan

Jenis Wavelet	Nilai <i>Threshold</i>	Citra Relevan	Citra Tidak Relevan	Rata-Rata Persentase Presisi
Haar	9.4	110	28	85,41%
	9.5	110	28	85,41%
	9.6	111	32	83,41%
	9.7	112	33	82,88%
Daub 4	27,9	133	121	58,37%
	28,0	133	121	58,37%
	28,1	133	122	57,62%
	28,2	133	122	57,62%
Sym 2	19,8	178	76	73,95%
	19,9	178	76	73,95%
	20,0	178	76	73,95%
	20,1	178	77	73,79%
Coif 1	22,5	134	103	63,03%
	22,6	135	104	62,81%
	22,7	136	104	62,98%
	22,8	136	106	62,62%

Hasil pengukuran dengan jarak Manhattan menunjukkan rata-rata persentase presisi terbesar untuk Wavelet Haar adalah 85,41% dengan nilai *threshold* 9,4, untuk Wavelet Daubechies 4 adalah 58,37% dengan nilai *threshold* 27,9, untuk Wavelet Symlets 2 adalah 73,95% dengan nilai *threshold* 19,8 dan untuk Wavelet Coiflets 1 adalah 63,03% dengan nilai *threshold* 22,5.

Hasil perbandingan presisi menunjukkan bahwa Wavelet Haar memperoleh rata-rata persentase presisi terbesar, yaitu

85,41%. Sedangkan Wavelet Daubechies 4 memperoleh rata-rata persentase presisi terkecil, yaitu 58,37%.

3. Pengujian dengan jarak Canberra

Hasil pengukuran kemiripan citra menggunakan jarak Canberra dengan nilai *threshold* dan jenis wavelet yang berbeda-beda ditunjukkan pada Tabel 5.9.

Tabel 5.9 Hasil Pengukuran Kemiripan Citra Menggunakan Jarak Canberra

Jenis Wavelet	Nilai <i>Threshold</i>	Citra Relevan	Citra Tidak Relevan	Rata-Rata Persentase Presisi
Haar	3,8	178	79	73,21%
	3,9	180	82	72,55%
	4,0	180	83	72,33%
	4,1	184	83	72,33%
Daub 4	6,2	149	99	65,03%
	6,3	150	99	65,33%
	6,4	152	101	65,26%
	6,5	153	102	65,35%
Sym 2	4,5	164	88	70,08%
	4,6	164	94	68,48%
	4,7	164	94	68,48%
	4,8	165	94	68,67%
Coif 1	5,5	154	95	66,42%
	5,6	154	96	66,14%
	5,7	155	96	66,14%
	5,8	155	96	66,14%

Hasil pengukuran dengan jarak Canberra menunjukkan rata-rata persentase presisi terbesar untuk Wavelet Haar adalah 73,21% dengan nilai *threshold* 3,8, untuk Wavelet Haar adalah 65,35% dengan nilai *threshold* 6,5, untuk Wavelet Symlets 2 adalah

70,08% dengan nilai *threshold* 4,5 dan untuk Wavelet Coiflets 1 adalah 66,42% dengan nilai *threshold* 5,5.

Hasil perbandingan presisi menunjukkan bahwa Wavelet Haar memperoleh rata-rata persentase presisi terbesar, yaitu 73,21%. Sedangkan Wavelet Daubechies 4 memperoleh rata-rata persentase presisi terkecil, yaitu 65,42%.

5.2 Pembahasan Hasil

Pembahasan hasil pengujian digunakan untuk mengetahui kinerja aplikasi temu kembali citra tenun menggunakan Transformasi Wavelet Diskrit. Berdasarkan hasil yang telah didapatkan pada subbab sebelumnya, aplikasi berhasil melakukan proses temu kembali citra tenun berdasarkan tingkat kesesuaian antara citra *query* dengan citra di *database*. Perhitungan persentase presisi dilakukan terhadap semua citra *query* dengan setiap jenis wavelet dan pengukuran kemiripan citra. Wavelet Haar menghasilkan rata-rata persentase presisi terbesar untuk setiap pengukuran kemiripan citra, yaitu sebesar 83,59% untuk metode jarak Euclidean, 85,41% untuk metode jarak Manhattan, dan 73,21% untuk metode jarak Canberra.

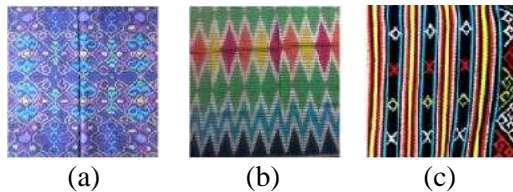
Selanjutnya, pengujian dilakukan untuk membandingkan hasil yang diperoleh dari setiap level proses dekomposisi. Pengujian dilakukan menggunakan Wavelet Haar dan metode jarak Manhattan.

Tabel 5.10 Perbandingan Presisi dari Dekomposisi Level 1 sampai Level 4

Level Dekomposisi	Nilai <i>Threshold</i> Terbaik	Citra Relevan	Citra Tidak Relevan	Rata-Rata Persentase Presisi
1	21,7	161	90	68,38%
2	17,8	146	79	72,74%
3	13,9	141	65	75,72%
4	9,4	110	28	85,41%

Hasil perbandingan presisi menunjukkan bahwa Wavelet Haar dengan dekomposisi level 4 memperoleh rata-rata persentase presisi terbesar, yaitu 85,41%. Sedangkan Wavelet Haar dengan dekomposisi level 1 memperoleh rata-rata persentase presisi terkecil, yaitu 68,38%.

Namun, tidak semua data citra dari setiap kelompok citra memiliki nilai presisi yang tinggi. Hal tersebut disebabkan oleh rendahnya kualitas citra yang digunakan. Kualitas citra yang rendah dapat terjadi karena proses pengambilan gambar dengan kamera yang kurang memperhatikan komposisi dan sudut pengambilan gambar. Sehingga, proses pengolahan citra menjadi tidak optimal. Contoh citra dengan kualitas rendah dapat dilihat pada Gambar 5.3.



Gambar 5.3 Contoh Citra dengan Kualitas Rendah.

- (a) Citra Tenun Bali;
- (b) Citra Tenun Jawa Tengah;
- (c) Citra Tenun NTT.

BAB VI PENUTUP

Bab ini berisi tentang simpulan-simpulan yang didapatkan berdasarkan penelitian yang telah dilakukan. Selain itu, bab ini juga berisi beberapa saran yang dapat digunakan untuk pengembangan penelitian.

6.1 Simpulan

Berdasarkan pengujian dan pembahasan terhadap hasil pengujian yang telah dilakukan terhadap aplikasi Temu Kembali Citra Tenun menggunakan Transformasi Wavelet Diskrit, maka dapat diperoleh beberapa simpulan sebagai berikut:

1. Proses temu kembali citra tenun menggunakan Transformasi Wavelet Diskrit sebagai metode ekstraksi dengan urutan proses *cropping* citra, *grayscale* citra, ekstraksi fitur tekstur citra dengan wavelet, dan proses pengukuran kemiripan citra dengan metode perhitungan jarak.
2. Metode Transformasi Wavelet Diskrit dengan jenis Wavelet Haar menghasilkan tingkat rata-rata persentase presisi tertinggi pada semua metode perhitungan jarak, yaitu sebesar 83,59% untuk metode jarak Euclidean, 85,41% untuk metode jarak Manhattan, dan 73,21% untuk metode jarak Canberra.
3. Tidak semua data citra dari setiap kelompok citra memiliki nilai presisi yang tinggi. Hal tersebut disebabkan oleh rendahnya kualitas citra yang digunakan.

6.2 Saran

Berdasarkan hasil yang dicapai pada penelitian ini, terdapat beberapa hal yang penulis kumpulkan sebagai saran untuk pengembangan selanjutnya, diantaranya adalah sebagai berikut:

1. Penambahan jenis motif tenun yang digunakan sehingga dapat memperkaya *database*.

2. Penambahan proses pada tahap pra-pemrosesan untuk memperbaiki kualitas citra sehingga dapat meningkatkan hasil dari tahap-tahap selanjutnya.
3. Pengembangan aplikasi Temu Kembali Citra Tenun yang terhubung langsung dengan kamera secara *realtime* dapat dikembangkan untuk mendukung konsep mesin pencari berbasis gambar ataupun konsep pemasaran *e-Commerce*.

DAFTAR PUSTAKA

- [1] A. Kr. Yadav, R. Roy, Vaishali, and A. P. Kumar, "Survey on Content Based Image Retrieval and Texture Analysis with Applications", *International Journal of Signal Processing, Image Processing and Pattern Recognition*, Vol. 7 (2014) 41-50.
- [2] M. Kokare, P. K. Biswas, and B. N. Chatterji, "Texture Image Retrieval Using Rotated Wavelet Filters", *Pattern Recognition Letters*, Vol. 28 Issue 10 (2007) 1240-1249.
- [3] N. M. Setiohardjo and A. Harjoko, "Analisis Tekstur untuk Klasifikasi Motif Kain (Studi Kasus Kain Tenun Nusa Tenggara Timur)", *IndoCEISS*, Vol. 8 (2014) No. 2.
- [4] S. Bharkad and M. Kokare, "Fingerprint Matching Using Discrete Wavelet Packet Transform", *IEEE Advance Computing Conference*, (2013) 1183-1188.
- [5] J. Samraj and NazreenBee, "Content Based Medical Image Retrieval Using Fuzzy C-Means Clustering With RF", *IJISSET*, Vol. 2 Issue 1 (2015, Jan) 512-518.
- [6] Cita Tenun Indonesia. (2016, Feb). Jenis Tenun. Available: http://www.tenunindonesia.com/jenis_tenun.php
- [7] A. Philip, B. Afolabi, A. Oluwaranti, and O. Oluwatolani, "Development of An Image Retrieval Model for Biomedical Image Database", *INTECH*, (2011) 311-329.
- [8] K. Kumar, "CBIR: Content Based Image Retrieval", *NCAIS*, (2010) 1-8.
- [9] S. Mallat, *A Wavelet Tour of Signal Processing*, 3rd ed., Burlington, MA (2009) Ch. 7.
- [10] R. C. Gonzales and R. E. Woods, *Digital Image Processing*, 2nd ed., Upper Saddle River, New Jersey (2001) Ch. 7.
- [11] G. Artigues, J. Portell, and A. G. Villafranca, "Discrete Wavelet Transform Fully Adaptive Prediction Error Coder: Image Data Compression based on CCSDS 122.0 and Fully Adaptive Prediction Error Coder", *SPIE Digital Library*, Vol. 7 Issue 2 (2013).

- [12] F. Wasilewski. (2016, Feb). Wavelet Browser [Online]. Available: <http://wavelets.pybytes.com>
- [13] P. V. N. Reddy and K. S. Prasad, "Multiwavelet Based Texture Features for Content Based Image Retrieval", *IJCST*, Vol. 2, Issue 1 (2011) 141-145.
- [14] A. M. Khan. "Fast Distance Metric Based Data Mining Techniques Using P-trees: k-Nearest-Neighbor Classification and k-Clustering", *ResearchGate*, (2001).
- [15] C. D. Manning, P. Raghavan, and H. Schutze. *An Introduction to Information Retrieval*, 1st ed., Cambridge (2009).

LAMPIRAN A

A.1 Kode Fungsi untuk Ekstraksi Fitur Tekstur Menggunakan Wavelet Haar

```
function waveletMoments =  
waveletHaarTransform(imGray)  
  
coeff_1 = dwt2(imGray, 'haar');  
coeff_2 = dwt2(coeff_1, 'haar');  
coeff_3 = dwt2(coeff_2, 'haar');  
coeff_4 = dwt2(coeff_3, 'haar');  
  
meanHaar = mean(coeff_4);  
stdHaar = std(coeff_4);  
  
waveletMoments = [meanHaar stdHaar]  
  
end
```

LAMPIRAN A (LANJUTAN)

A.2 Kode Fungsi untuk Ekstraksi Fitur Tekstur Menggunakan Wavelet Daubechies 4

```
function waveletMoments =  
waveletDaubechiesTransform(imGray)  
  
coeff_1 = dwt2(imGray, 'db4');  
coeff_2 = dwt2(coeff_1, 'db4');  
coeff_3 = dwt2(coeff_2, 'db4');  
coeff_4 = dwt2(coeff_3, 'db4');  
  
meanDb = mean(coeff_4);  
stdDb = std(coeff_4);  
  
waveletMoments = [mean stdDb]  
  
end
```

LAMPIRAN A (LANJUTAN)

A.3 Kode Fungsi untuk Ekstraksi Fitur Tekstur Menggunakan Wavelet Symlets 2

```
function waveletMoments =  
waveletSymletsTransform(imGray)  
  
coeff_1 = dwt2(imGray, 'sym2');  
coeff_2 = dwt2(coeff_1, 'sym2');  
coeff_3 = dwt2(coeff_2, 'sym2');  
coeff_4 = dwt2(coeff_3, 'sym2');  
  
meanSym = mean(coeff_4);  
stdSym = std(coeff_4);  
  
waveletMoments = [meanSym stdSym]  
  
end
```

LAMPIRAN A (LANJUTAN)

A.4 Kode Fungsi untuk Ekstraksi Fitur Tekstur Menggunakan Wavelet Coiflets 1

```
function waveletMoments =  
waveletCoifletsTransform(imGray)  
  
coeff_1 = dwt2(imGray, 'coif1');  
coeff_2 = dwt2(coeff_1, 'coif1');  
coeff_3 = dwt2(coeff_2, 'coif1');  
coeff_4 = dwt2(coeff_3, 'coif1');  
  
meanCoif = mean(coeff_4);  
stdCoif = std(coeff_4);  
  
waveletMoments = [meanCoif stdCoif]  
  
end
```

LAMPIRAN B

B1. Kode Fungsi Pengukuran Kemiripan Citra dengan Mengatur Jumlah Citra untuk Jarak Euclidean

```
function Euclidean(numOfReturnedImages,
queryImageFeatureVector, dataset)

global sortedEuclidImgs

query_img_name = queryImageFeatureVector(:,
end);
dataset_img_names = dataset(:, end);

queryImageFeatureVector(:, end) = [];
dataset(:, end) = [];

euclidean = zeros(size(dataset, 1), 1);

for k = 1:size(dataset, 1)
    euclidean(k) = sqrt( sum( power( dataset(k,
:) - queryImageFeatureVector, 2 ) ) );
end

euclidean = [euclidean dataset_img_names];

[sortedEuclidDist indxs] = sortrows(euclidean);
indxsortedEuclidImgs = sortedEuclidDist(:, 2)
sortedEuclidImgs = sortedEuclidDist(:,1)

arrayfun(@cla, findall(0, 'type', 'axes'));

str_name = int2str(query_img_name);
query_img = imread( strcat('query\'', str_name,
'.jpg') );
subplot(3, 4, 1);
imshow(query_img, []);
title('Citra Query', 'Color', [0 0 0]);
```


LAMPIRAN B (LANJUTAN)

```
for m = 1:numOfReturnedImages
    img_name = indxsortedEuclidImgs(m);
    img_name = int2str(img_name);
    str_img_name = strcat('images\', img_name,
        '.jpg');
    returned_img = imread(str_img_name);
    subplot(3, 4, m+1);
    imshow(returned_img, []);
end

end
```

LAMPIRAN B (LANJUTAN)

B2. Kode Fungsi Pengukuran Kemiripan Citra dengan Mengatur Jumlah Citra untuk Jarak Manhattan

```
function Manhattan(numOfReturnedImages,
queryImageFeatureVector, dataset)

global sortedManhatImgs

query_image_name = queryImageFeatureVector(:,
end);
dataset_image_names = dataset(:, end);

queryImageFeatureVector(:, end) = [];
dataset(:, end) = [];

manhattan = zeros(size(dataset, 1), 1);

for k = 1:size(dataset, 1)
    manhattan(k) = sum( abs(dataset(k, :) -
        queryImageFeatureVector) );
end

manhattan = [manhattan dataset_image_names];

[sortedManhatDist indx] = sortrows(manhattan);
indxsortedManhatImgs = sortedManhatDist(:,2);
sortedManhatImgs = sortedManhatDist(:,1)

arrayfun(@cla, findall(0, 'type', 'axes'));

str_name = int2str(query_image_name);
queryImage = imread( strcat('query\'', str_name,
'.jpg') );
subplot(3, 4, 1);
imshow(queryImage, []);
title('Citra Query', 'Color', [1 0 0]);
```

LAMPIRAN B (LANJUTAN)

```
for m = 1:numOfReturnedImages
    img_name = indxsortedManhatImgs(m);
    img_name = int2str(img_name);
    str_name = strcat('images\', img_name,
        '.jpg');
    returnedImage = imread(str_name);
    subplot(3, 4, m+1);
    imshow(returnedImage, []);
end

end
```

LAMPIRAN B (LANJUTAN)

B3. Kode Fungsi Pengukuran Kemiripan Citra dengan Mengatur Jumlah Citra untuk Jarak Canberra

```
function Canberra(numOfReturnedImages,
queryImageFeatureVector, dataset)

global sortedCanbImgs

query_im_name = queryImageFeatureVector(:, end);
dataset_im_names = dataset(:, end);

queryImageFeatureVector(:, end) = [];
dataset(:, end) = [];

canberra = zeros(length(dataset), 1);

for k = 1:length(dataset)
    canberra(k) = sum ((abs( dataset(k, :) -
        queryImageFeatureVector )) ./ (abs
        (dataset(k, :)) + abs
        (queryImageFeatureVector))) ;
end

canberra = [canberra dataset_im_names];

[sortCanbDist indxs] = sortrows(canberra);
indxsortedCanbImgs = sortCanbDist(:, 2)
sortedCanbImgs = sortCanbDist(:,1)

arrayfun(@cla, findall(0, 'type', 'axes'));

str_img_name = int2str(query_im_name);
query_im = imread( strcat('query\'',
str_img_name, '.jpg') );
subplot(3, 4, 1);
imshow(query_im, []);
```

LAMPIRAN B (LANJUTAN)

```
title('Citra Query', 'Color', [1 0 0]);

for m = 1:numOfReturnedImages
    im_name = indxsortedCanbImgs(m);
    im_name = int2str(im_name);
    str_im_name = strcat('images\', im_name,
        '.jpg');
    returned_im = imread(str_im_name);
    subplot(3, 4, m+1);
    imshow(returned_im, []);
end

end
```

LAMPIRAN B (LANJUTAN)

B4. Kode Fungsi Pengukuran Kemiripan Citra dengan Mengatur Nilai *Threshold* untuk Jarak Euclidean

```
function Euclidean2(queryImageFeatureVector,
dataset)

global sortedEuclidImgs thres

query_img_name = queryImageFeatureVector(:,
end);
dataset_img_names = dataset(:, end);

queryImageFeatureVector(:, end) = [];
dataset(:, end) = [];

euclidean = zeros(size(dataset, 1), 1);

for k = 1:size(dataset, 1)
    euclidean(k) = sqrt( sum( power( dataset(k,
        :) - queryImageFeatureVector, 2 ) ) );
end

euclidean = [euclidean dataset_img_names];

[sortedEuclidDist indxs] = sortrows(euclidean);
indxsortedEuclidImgs = sortedEuclidDist(:, 2);
sortedEuclidImgs = sortedEuclidDist(:,1)

arrayfun(@cla, findall(0, 'type', 'axes'));

str_name = int2str(query_img_name);
query_img = imread( strcat('query\ ', str_name,
'.jpg') );
subplot(3, 4, 1);
imshow(query_img, []);
```

LAMPIRAN B (LANJUTAN)

```

title('Citra Query', 'Color', [0 0 0]);

numOfReturnedImages=0;
i=1;
while sortedEuclidImgs(i) <= thres

    numOfReturnedImages=numOfReturnedImages+1;
    i=i+1;
end

for m = 1:numOfReturnedImages
    img_name = indxsortedEuclidImgs(m);
    img_name = int2str(img_name);
    str_img_name = strcat('images\'', img_name,
        '.jpg');

    returned_img = imread(str_img_name);
    subplot(3, 4, m+1);
    imshow(returned_img, []);
end

end

```

LAMPIRAN B (LANJUTAN)

B5. Kode Fungsi Pengukuran Kemiripan Citra dengan Mengatur Nilai *Threshold* untuk Jarak Manhattan

```
function Manhattan2(queryImageFeatureVector,
dataset)

global sortedManhatImgs thres

query_image_name = queryImageFeatureVector(:,
end);
dataset_image_names = dataset(:, end);

queryImageFeatureVector(:, end) = [];
dataset(:, end) = [];

manhattan = zeros(size(dataset, 1), 1);

for k = 1:size(dataset, 1)
    manhattan(k) = sum( abs(dataset(k, :) -
        queryImageFeatureVector) );
end

manhattan = [manhattan dataset_image_names];

[sortedManhatDist indx] = sortrows(manhattan);
indxsortedManhatImgs = sortedManhatDist(:,2);
sortedManhatImgs = sortedManhatDist(:,1)

arrayfun(@cla, findall(0, 'type', 'axes'));

str_name = int2str(query_image_name);
queryImage = imread( strcat('query\ ', str_name,
'.jpg') );
subplot(3, 4, 1);
imshow(queryImage, []);
```


LAMPIRAN B (LANJUTAN)

```

title('Citra Query', 'Color', [0 0 0]);

numOfReturnedImages=0;
i=1;
while sortedManhatImgs(i) <= thres

    numOfReturnedImages=numOfReturnedImages+1;
    i=i+1;
end

for m = 1:numOfReturnedImages
    img_name = indxsortedManhatImgs(m);
    img_name = int2str(img_name);
    str_name = strcat('images\'', img_name,
        '.jpg');

    returnedImage = imread(str_name);
    subplot(3, 4, m+1);
    imshow(returnedImage, []);
end

end

```

LAMPIRAN B (LANJUTAN)

B6. Kode Fungsi Pengukuran Kemiripan Citra dengan Mengatur Nilai *Threshold* untuk Jarak Canberra

```
function Canberra2(queryImageFeatureVector,
dataset)

global sortedCanbImgs thres

query_im_name = queryImageFeatureVector(:, end);
dataset_im_names = dataset(:, end);

queryImageFeatureVector(:, end) = [];
dataset(:, end) = [];

canberra = zeros(length(dataset), 1);
for k = 1:length(dataset)
    canberra(k) = sum ((abs( dataset(k, :) -
        queryImageFeatureVector )) ./ (abs
        (dataset(k, :)) + abs
        (queryImageFeatureVector))) ;
end

canberra = [canberra dataset_im_names];

[sortCanbDist indxs] = sortrows(canberra);
indxsortedCanbImgs = sortCanbDist(:, 2);
sortedCanbImgs = sortCanbDist(:,1)

arrayfun(@cla, findall(0, 'type', 'axes'));

str_img_name = int2str(query_im_name);
query_im = imread( strcat('query\'',
str_img_name, '.jpg') );
subplot(3, 4, 1);
imshow(query_im, []);
```

LAMPIRAN B (LANJUTAN)

```

title('Citra Query', 'Color', [1 0 0]);

numOfReturnedImages=0;
i=1;
while sortedCanbImgs(i) <= thres

    numOfReturnedImages=numOfReturnedImages+1;
    i=i+1;
end

for m = 1:numOfReturnedImages
    im_name = indxsortedCanbImgs(m);
    im_name = int2str(im_name);
    str_im_name = strcat('images\'', im_name,
        '.jpg');

    returned_im = imread(str_im_name);
    subplot(3, 4, m+1);
    imshow(returned_im, []);
end

end

```

BIODATA PENULIS



Penulis bernama lengkap Agista Regiaswuri, yang biasa dipanggil Gista. Penulis dilahirkan di Madiun pada tanggal 12 Agustus 1994 dan bertempat tinggal di Desa Klumutan, Kecamatan Saradan, Kabupaten Madiun. Penulis menempuh pendidikan di SDN Klumutan 01 pada tahun 2000-2006, kemudian melanjutkan di SMPN 1 Mejayan pada tahun 2006-2009, dan SMAN 1 Mejayan pada tahun 2009-2012. Tahun

2012, penulis diterima di jurusan Matematika ITS melalui jalur SNMPTN Undangan.

Semenjak kuliah, penulis tertarik untuk mempelajari bahasa pemrograman. Bahasa Pemrograman yang pernah penulis pelajari adalah C, C++, Java, dan PHPMySQL. Oleh karena itu, penulis memilih bidang minat Ilmu Komputer.

Semasa menempuh jenjang pendidikan S1, penulis juga mengikuti beberapa organisasi kemahasiswaan ITS, diantaranya adalah Himpunan Mahasiswa Matematika (HIMATIKA) ITS dan UKM Fotografi ITS. Untuk mendapatkan informasi yang berhubungan dengan Tugas Akhir ini dapat ditujukan ke alamat email : aregiaswuri@gmail.com.